Methods for Constructing an Opinion Network for Politically Controversial Topics

Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
der Naturwissenschaftlich-Technischen Fakultät I
der Universität des Saarlandes

Rawia Awadallah
Max-Planck-Institut für Informatik

Saarbrücken
2012

Dekan der Naturwissenschaftlich-Technischen Fakultät I	Prof. Dr.-Ing. Mark Groves
Vorsitzender der Prüfungskommission	Prof. Dr. Dietrich Klakow
Berichterstatter	Prof. Dr.-Ing. Gerhard Weikum
Berichterstatter	Prof. Dr. Maya Ramanath
Berichterstatter	Prof. Dr.-Ing. Andreas Rauber
Beisitzer	Dr.-Ing. Ralf Schenkel
Tag des Promotionskollquiums	21.12.2012

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe.

Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Saarbrücken, den 31.10.2012

(Rawia Awadallah)

Rawia Awadallah

To my husband, Dr. Anwar Awadallah

Acknowledgment

I would like to express my deepest gratitude to my advisor, Prof. Dr.-Ing. Gerhard Weikum. Without his scientific guidance, insightful comments, exemplary leadership, and more importantly his extreme patience with me, this work would not have been possible.

I would like also to thank Prof. Dr.-Ing. Andreas Rauber for reviewing the thesis. I first learnt how to do proper research with him and I believe our work together during my master thesis gave me a first-hand experience on how to carry out scientific research.

I am very thankful to my co-advisor Dr. Maya Ramanath. Her support, creative ideas, and readiness to lend me a helping hand are highly appreciated.

I thank all my friends at MPI for the memorable moments we enjoyed together. In particular I warmly thank Dimiter Denev, Katja Hose, Gerard de Melo, Iris Miliaraki, Mohamed Yahya, and Yafang Wang, who were always very generous to help me, and with whom I made great memories. I also thank Ndapandula Nakashole for the nice time we spent together at MPI, and especially for her smile every morning when she used to pass by my office. Very especial thank to Bilyana Taneva for being a very close friend to me. She used to listen to my problems, and to discuss with me my work ideas friendly and patiently.

I would like to thank my parents and my family. In particular I warmly thank my daughters Haneen, Hala, Dima, and Nalah who had to suffer with their busy and stressed mother. Last but not least, I can not find enough words to express my thankful and appreciation to my husband Anwar who will always be my best friend. He supports me morally, emotionally, and is always there in the good times and in the bad times. Coming to Saarbrücken, being able to continue my PhD studies, overcoming the stressful and disappointing times, and coping with four children, could not have been possible without him.

Abstract

The US presidential race, the re-election of President Hugo Chavez, and the economic crisis in Greece and other European countries are some of the controversial topics being played on the news everyday. To understand the landscape of opinions on political controversies, it would be helpful to know which politician or other stakeholder takes which position - support or opposition - on specific aspects of these topics. The work described in this thesis aims to automatically derive a map of the opinions-people network from news and other Web documents. The focus is on acquiring opinions held by various stakeholders on politically controversial topics. This opinions-people network serves as a knowledge-base of opinions in the form of ⟨`opinion holder`⟩ ⟨`opinion`⟩ ⟨`topic`⟩ triples. Our system to build this knowledge-base makes use of online news sources in order to extract opinions from text snippets. These sources come with a set of unique challenges. For example, processing text snippets involves not just identifying the topic and the opinion, but also attributing that opinion to a specific opinion holder. This requires making use of deep parsing and analyzing the parse tree. Moreover, in order to ensure uniformity, both the topic as well the opinion holder should be mapped to canonical strings, and the topics should also be organized into a hierarchy. Our system relies on two main components: i) *acquiring opinions* which uses a combination of techniques to extract opinions from online news sources, and ii) *organizing topics* which crawls and extracts debates from online sources, and organizes these debates in a hierarchy of political controversial topics. We present systematic evaluations of the different components of our system, and show their high accuracies. We also present some of the different kinds of applications that require political analysis. We present some application requires political analysis such as identifying flip-floppers, political bias, and dissenters. Such applications can make use of the knowledge-base of opinions.

Kurzfassung

Kontroverse Themen wie das US-Präsidentschaftsrennen, die Wiederwahl von Präsident Hugo Chavez, die Wirtschaftskrise in Griechenland sowie in anderen europäischen Ländern werden täglich in den Nachrichten diskutiert. Um die Bandbreite verschiedener Meinungen zu politischen Kontroversen zu verstehen, ist es hilfreich herauszufinden, welcher Politiker bzw. Interessenvertreter welchen Standpunkt (Pro oder Contra) bezüglich spezifischer Aspekte dieser Themen einnimmt. Diese Dissertation beschreibt ein Verfahren, welches automatisch eine Übersicht des Meinung-Mensch Netzwerks aus aktuellen Nachrichten und anderen Web-Dokumenten ableitet. Der Fokus liegt hierbei auf dem Erfassen von Meinungen verschiedener Interessenvertreter bezüglich politisch kontroverser Themen. Dieses Meinung-Mensch-Netzwerk dient als Wissensbasis von Meinungen in Form von Tripeln: ⟨`Meinungsvertreter`⟩ ⟨`Meinung`⟩ ⟨`Thema`⟩. Um diese Wissensbasis aufzubauen, nutzt unser System Online-Nachrichten und extrahiert Meinungen aus Textausschnitten. Quellen von Online-Nachrichten stellen eine Reihe von besonderen Anforderungen an unser System. Zum Beispiel umfasst die Verarbeitung von Textausschnitten nicht nur die Identifikation des Themas und der geschilderten Meinung, sondern auch die Zuordnung der Stellungnahme zu einem spezifischen Meinungsvertreter. Dies erfordert eine tiefgründige Analyse sowie eine genaue Untersuchung des Syntaxbaumes. Um die Einheitlichkeit zu gewährleisten, müssen darüber hinaus Thema sowie Meinungsvertreter auf ein kanonisches Format abgebildet und die Themen hierarchisch angeordnet werden. Unser System beruht im Wesentlichen auf zwei Komponenten: i) *Erkennen von Meinungen*, welches verschiedene Techniken zur Extraktion von Meinungen aus Online-Nachrichten beinhaltet, und ii) *Erkennen von Beziehungen zwischen Themen*, welches das Crawling und Extrahieren von Debatten aus Online-Quellen sowie das Organisieren dieser Debatten in einer Hierarchie von politisch kontroversen Themen umfasst. Wir präsentieren eine systematische Evaluierung der verschiedenen Systemkomponenten, welche die hohe Genauigkeit der von

uns entwickelten Techniken zeigt. Wir diskutieren außerdem verschiedene Arten von Anwendungen, die eine politische Analyse erfordern, wie zum Beispiel die Erkennung von Opportunisten, politische Voreingenommenheit und Dissidenten. All diese Anwendungen können durch die Wissensbasis von Meinungen umfangreich profitieren.

Summary

The political controversial topics being played and discussed in great depth on the news everyday such as "Greece bailout" and "Wikileaks release" have many different facets including "Ecuador grant of asylum to WikiLeaks' Assange". These topics have also many stakeholders including nations such as "Greece", "Ecuador", etc., or people such as "Angela Merkel", "Julian Assange", etc..

In order to understand, navigate and analyze this complicated landscape of issues and opinions taken by various stakeholders, it would be helpful if there were a browseable and queryable ***opinion-base***. Such an opinion-base would organize controversial topics according to their various, more fine-grained facets and would extract and store the opinions of people on these facets and information about who expressed an opinion on one or more topics and supported or opposed a certain direction. This kind of network could then be used to perform different kinds of analyses, such as, "Who originally supported Greece bailout, but now changed their minds?", "Which news media support Assange?", etc.. Such analyses are typically based on aggregating many statements and work well for coarse-grained topics. However, political analysts are often interested in individual and brief statements, as reported or quoted in news media, and their ***pro/con polarity*** with regard to ***fine-grained topics*** such as "deporting illegal immigrants", or "banning Facebook at workplaces". The outcome of analyzing these inputs should be a crisp set of structured records of the form: ⟨`opinion holder`⟩ ⟨`opinion`⟩ ⟨`opinion target`⟩. For example, ⟨`Angela Merkel`⟩ ⟨`support`⟩ ⟨`Greece bailout`⟩. This desired output involves aggregating statements, but only for the same pair of opinion holder and topic facet. The overall set of such records forms the opinion network.

The challenge in building the envisioned opinion network lies in the fact that the input is merely natural-language text, such as news articles or social media, where it is difficult to spot phrases that denote individual politicians or controversial topics and map them into a canonical representation. There are many

forms of expressing opinions in newspapers, broadcast stations, online forums, and all kinds of social media. Acquiring political opinions involves a number of challenging tasks. This thesis presents these tasks and solutions to key aspects of them.

- **Acquiring opinion sentences:** Opinionated sentences of politicians or other stakeholders on political controversies from news articles are to be acquired. These sentences usually comes in short texts, so whatever linguistic features are used tend to be sparse. Moreover, they are reported either in a direct way by explicitly stating the opinion holders' opinions, or in an indirect way in which the beliefs or the arguments of the opinion holders implicitly state the opinions.

- **Identifying opinion holders:** From the acquired opinion sentences, entities expressing opinions are to be identified.

- **Identifying opinion polarities:** From the acquired opinion sentences, the polarities of the opinions (positive or negative, pro or con) are to be identified. These polarities are not always explicitly mentioned in news.

- **Identifying opinion targets:** From the acquired opinion sentences, the topics on which opinions are expressed are to be identified.

- **Political controversial nature:** The acquired topics or their facets should be *politically controversial*.

- **Canonicalization:** Since each facet can be expressed in different ways, they should be *canonicalized*.

- **Fine granularity:** Once we have canonical forms of facets, their topics should be identified. Moreover, these topics are to be organized into a hierarchy of topics.

- **Topic-dependent polarity:** The same opinion sentence can have different polarities for different fine-grained topics, and there is a need to identify the polarity on each fine-grained topic.

Our system to build this knowledge-base makes use of online news sources in order to extract opinions from text snippets. The system relies on two main

components: i) *acquiring opinions* which uses a combination of techniques to extract opinions from online news sources, and ii) *organizing topics* which crawls and extracts debates from online sources, and organizes these debates in a hierarchy of political controversial topics. We present systematic evaluations of the different components of our system, and show their high accuracies. We also present some of the different kinds of applications built on top of the opinions knowledge-base which require political analysis such as identifying flip-floppers, political bias, and dissenters.

Zusammenfassung

In den täglichen Nachrichten werden politisch kontroverse Themen teilweise sehr detailliert diskutiert, so zum Beispiel die “Griechenlandkrise” und “Wikileaks-Veröffentlichungen”. Diese Theman jedoch haben viele verschiedene Facetten, wie zum Beispiel “Ecuador gewährt Asyl für Wikileaks-Gründer Assange”, und ebenso diverse Vertreter; darunter Nationen wie “Griechenland” und “Ecuador” aber auch Personen wie “Angela Merkel” und “Julian Assange”.

Eine große Hilfe zur Bewältigung (Verstehen, Navigation, Analyse etc.) dieses komplizierten Spektrums von Themen und Meinungen verschiedener Vertreter wäre eine *Wissensbasis*, die konkrete Anfragen beantwortet aber auch einfaches Durchsuchen mit Hilfe eines Browsers ermöglicht. Eine derartige Wissensbasis von Meinungen sollte darüber hinaus kontroverse Themen mit ihren zahlreichen Facetten und wechselseitigen Beziehungen erfassen. Zusätzlich sollten auch Meinungen verschiedener Personen extrahiert, gespeichert und zusammen mit den Themen, auf die sie sich beziehen, sowie ihrer Position (Pro oder contra) erfasst werden. Dieses Netzwerk könnte dann als Basis für verschiedene Analysen dienen, zum Beispiel: “Wer unterstützte zunächst den Rettungsschirm für Griechenland, hat aber später seiner Meinung geändert?” und “Welche Medien unterstützen Assange?”. Gewöhnliche Analysen bauen in der Regel auf der Aggregation vieler Aussagen auf und liefern gute Ergebnisse für grobe thematische Kategorien. Allerdings sind für politische Analysen oftmals einzelne kurze Aussagen, wie sie in den Nachrichten wiedergegeben und zitiert werden, sowie deren Polarität (Pro oder Contra) hinsichtlich spezifischerer Themen interessant, wie zum Beispiel “Abschiebung illegaler Einwanderer” oder “Verbot von Facebook am Arbeitsplatz”. Das Ergebnis der Analyse sollte eine klare strukturierte Menge von Datensätzen folgender Form sein: ⟨`Meinungsvertreter`⟩ ⟨`Meinung`⟩ ⟨`Thema`⟩. Zum Beispiel: ⟨`Angela Merkel`⟩ ⟨`Unterstützung`⟩ ⟨`Rettungsschirm für Griechenland`⟩. Ein derartiges Ergebnis beinhaltet die Aggregation von Aussagen bezüglich Paare aus Meinungsvertreter und thematischer Facette. Die

Gesamtmenge dieser Datensätze bildet ein Meinungsnetzwerk.

Die besondere Herausforderung bei der Realisierung dieses Meinungsnetzwerks ist in der Tatsache begründet, dass die Grundlage lediglich natürlichsprachige Texte aus Nachrichten oder sozialen Netzwerken sind, was die Erkennung einzelner Aussagen, die Politiker oder kontroverse Themen eindeutig bezeichnen, sowie die Abbildung in ein kanonisches Format zu besonderen Aufgaben macht. Es gibt viele verschiedene Formen der Meinungsäußerung in den Nachrichten, Radiostationen, Online-Foren und allen Arten sozialer Medien. Die Erkennung und Extraktion politischer Meinungen beinhaltet eine Reihe von anspruchsvollen Aufgaben. Hauptgegenstand dieser Arbeit ist daher die Diskussion dieser Aufgaben sowie die Präsentation geeigneter Algorithmen.

- **Erkennen von Meinungen**: Aussagen, die Meinungen von Politikern oder anderen Interessenvertretern bezüglich politisch kontroverse Themen darstellen, müssen zunächst erkannt werden. Derartige Aussagen kommen üblicherweise in Form von kurzen Texten vor, so dass linguistische Merkmale selten zur Auswertung herangezogen werden können. Meinungen werden dabei entweder direkt ausgedrückt, d.h. durch explizite Darlegung, oder indirekt, d.h. die Glaubensgrundsätze bzw. die Argumente des Meinungsvertreters drücken implizit seine Meinung aus.

- **Identifikation des Meinungsvertreters**: Aussagen, die eine Meinung ausdrücken, müssen dem zugehörigen Meinungsvertreter zugeordnet werden.

- **Identifikation der Polarität**: Zu jeder Aussage, die eine Meinung ausdrückt, muss die Polarität (positiv oder negativ bzw. Pro oder Contra) identifiziert werden. Diese wird jedoch nicht immer explizit in den Nachrichten erwähnt.

- **Identifikation des Themas**: Auch die Themen, über die eine Meinung ausgedrückt wurde, müssen auf Basis der Aussagen identifiziert werden.

- **Politische Brisanz**: Die identifizierten Themen bzw. ihre Facetten sollten *politisch umstritten* sein.

- **Formalisierung**: Da jede Facette auf unterschiedliche Weise ausgedrückt werden kann, sollte die Facette in ein *kanonisches Format* überführt werden.

- **Detailgenauigkeit**: Nachdem die Facetten in ein kanonisches Format überführt wurden, sollten sie übergeordneten Themen zugeordnet werden, welche wiederum in einem hierarchischen Geflecht in Beziehung zueinander stehen.

- **Themenabhängige Polarität**: In ein und derselben Aussage können verschiedene detaillierte Themen mit unterschiedlicher Polarität enthalten sein, so dass die Polarität für jedes Thema separat analysiert werden muss.

Unser System beruht im Wesentlichen auf zwei Komponenten: i) *Erkennen von Meinungen*, welches verschiedene Techniken zur Extraktion von Meinungen aus Online-Nachrichten beinhaltet und ii) *Erkennen von Beziehungen zwischen Themen*, welches das Crawling und Extrahieren von Debatten aus Online-Quellen sowie das Organisieren dieser Debatten in einer Hierarchie von politisch kontroversen Themen umfasst. Wir präsentieren eine systematische Evaluierung der verschiedenen Systemkomponenten, welche die hohe Genauigkeit der von uns entwickelten Techniken zeigt. Wir diskutieren außerdem verschiedene Arten von Anwendungen, die eine politische Analyse erfordern, wie zum Beispiel die Erkennung von Opportunisten, politische Voreingenommenheit und Dissidenten. All diese Anwendungen können durch die Wissensbasis von Meinungen umfangreich profitieren.

Contents

Chapter 1

Introduction

Textual information in the world expresses facts and opinions. Facts are objective statements about entities, events and their properties. Opinions are subjective statements that describe people's sentiments, appraisals or feelings toward entities, events and their properties. Conflicting opinions exist in a diverse range of domains, including science, politics, entertainments, etc.. The analysis of what other people think has a direct impact on people's understanding, interpretation, and decision-making activities. The "crises in Libya and Syria", the debates about the "economic crisis in Greece", and "the downrating of the USA's creditworthiness" are some of the *political controversial topics* being played on the news everyday and directly affect the lives of many people.

Existing research on textual analysis used to focus on the retrieval and analysis of factual information. The lack of studies on opinions was because of the lack of opinionated text before the Web 2.0 era. Now, opinions can be found almost everywhere: blogs, social networking sites like Facebook and Twitter, news portals, e-commerce sites, etc.. The large number of diverse sources of opinions makes it possible to find out about the experiences of people and their opinions, positive or negative, regarding any product, service or topic. However, it is difficult for a human reader to find relevant sources, extract relevant sentences with opinions, read them, summarize them, and organize them into usable forms. Thus, automated opinion discovery and summarization systems are needed. Opinion mining grows out of this need. It is a challenging problem in natural language processing and text mining.

In this chapter, we first give a brief introduction about opinion ming in Section 1.1. In Section 1.2, we present the research problems addressed in this thesis.

Our contributions are described in Section 1.3. Finally the thesis outline is given in Section 1.4.

1.1 Opinion Mining

Opinion mining is a richly researched topic that gained a lot of attention in recent years for business intelligence, smart advertisements, marketing campaigns, etc.. It is the computational study of people's opinions, appraisals, and emotions toward entities, events and their attributes [94, 81]. The research in the field focuses on *subjectivity classification* and *polarity classification*.

Subjectivity classification aims to determine whether a sentence is subjective or objective [129]. *Polarity classification* which is also known as *sentiment classification*, aims to detect whether an opinionated document or sentence expresses a positive or negative opinion [94]. The research in the field also addresses the issue of identifying the *topics* (e.g., abortion) and their *facets* (e.g., late-term abortion), or the *objects* (e.g., camera) and their *features* (e.g., picture quality) on which a sentiment is expressed. Applications of opinion mining can range from polarity categorization in reviews, to determining the strength of opinions in news articles, to identifying perspectives in political debates to analyzing mood in blogs. Typically, these cues are aggregated to form opinion profiles, based on many opinions like customer reviews or postings in discussion forums.

In this section, we refer to three main surveys on opinion mining [94, 81, 67], in order to sketch the state of the art.

1.1.1 Key Concepts

Researchers have considered a wide range of problems over a variety of different types of corpora. We present key concepts involved in these problems.

1. **Sentence subjectivity:** An ***objective sentence*** expresses some factual information about the world, while a ***subjective sentence*** expresses some personal beliefs or feelings.

2. **Object and feature:** The term ***object*** denotes the *opinion target* entity that an opinion is expressed on. It can be a product, a service, an individual, an

organization, a topic, or an event. If an ***object*** represents a product, it can have a set of ***features***. Each ***feature*** may have its own ***sub-features***. Thus, an ***object*** can be hierarchically decomposed based on the part-of relation. If the object represents a ***topic***, it can have a set of ***sub-topics***, which can also have its own ***sub-sub-topics*** and so on. The finest-grained topics are called ***facets***.

3. **Opinion holder:** The holder of an opinion which is also called the ***opinion source*** represents a person, an organization, a country, etc., that expresses the opinion. In the case of product reviews and blogs, opinion holders are usually the authors of the posts. Opinion holders are more important in news articles because they often explicitly state them.

4. **Opinion and polarity:** An ***opinion*** is a positive or a negative view or appraisal on an object or feature by an opinion holder. A ***positive/negative opinion*** is called ***opinion polarity***.

1.1.2 Subjectivity Classification

This task addresses the problem of deciding whether a given document contains subjective information or not, and identifying which portions of the document are subjective. This problem was the focus of the 2006 Blog track at TREC.

Early work examined the effects of adjectives on sentence subjectivity [129]. The goal was to tell whether a given sentence is subjective or not judging from the adjectives appearing in that sentence. The work in [128] presents a comprehensive survey of subjectivity recognition using different cues and features. Many research works used supervised learning algorithms [127, 129, 135]. For example, [127] performs subjectivity classification using a naïve Bayesian classifier. Other approaches aims to save the manual labeling effort required by supervised methods. For example a bootstrapping approach to label training data automatically is reported in [106]. A two high precision classifiers are used to automatically identify subjective and objective sentences. The two classifiers use lists of lexical items that are strong subjectivity cues. The extracted sentences are then added to the training data to learn patterns. The patterns are then used to automatically identify more subjective and objective sentences, and are then added to the training set, for the next iteration of the algorithm. For pattern learning, a set of syntactic templates are provided to restrict the kinds of patterns to be learned.

1.1.3 Polarity Classification

Many works on polarity-related classification/regression/ranking address the problem with the following general character: given an opinionated piece of text, classify the opinion as falling under one of two opposing polarities. The binary classification task of labeling an opinionated text as expressing either an overall positive or an overall negative opinion is called *polarity classification*. Polarity classification works on two levels: document level, and sentence level.

1.1.3.1 Document-Level Polarity Classification

Given a documents, a classifier determines whether the document expresses a positive or a negative opinion on an object. The classification approaches that consider this level of granularity assume that the opinionated document expresses opinions on a single object and the opinions are from a single opinion holder. This assumption holds for customer reviews of products and services. However, it may not hold for a forum for blog posting since authors may express opinions on multiple objects. Document-level polarity classification can be based on ***supervised learning***, or on ***unsupervised methods***.

Supervised learning methods can be readily applied to polarity classification, (e.g., Bayesian methods or, support vector machines (SVM)). For example, [95] classifies movie reviews into two classes, positive and negative. It was shown that using unigrams as features in classification performed well with either naïve Bayesian or SVM. Subsequent research used more kinds of features and learning techniques such as:

1. Words and their frequency, or the Tf–idf weighting scheme from information retrieval,

2. Part-of-speech tags. For example adjectives are important indicators of subjectivities and opinions,

3. Opinion words that indicate positive or negative opinions. For example, “beautiful”, “wonderful”, “good”, and “amazing” are positive opinion

words, while “bad”, and “poor” are negative opinion words,

4. Syntactic dependencies generated from deep parsing,

5. Negation. Since their appearances often change the opinion polarity.

Other works aim to determine the author’s evaluation with respect to a multi-point scale (e.g., one to five “stars” for a review [92]). The problem is formulated as a ***regression problem*** since ratings are ordinal. Labeled data from one domain and unlabeled data from the target domain and general opinion words have been used as features for adaptation [23].

The method in [122] performs review classification using an ***unsupervised technique***. First, it extracts phrases containing adjectives or adverbs since they are good indicators of subjectivity and opinions. To determine the polarity of the extracted phrases, their method considers two consecutive words, where one word of the pair is an adjective/adverb and the other is a context word (e.g., nice picture) , if their POS tags conform to any of a predefined set of patterns. Second, the method estimates the polarity of the adjective/adverb word based on its mutual association with the positive reference word “excellent” and its association with the negative reference word “poor”. The probabilities are calculated based on the numbers of hits returned by the search engine for the queries generated from the reference words and the adjective/adverb words. Finally, the algorithm computes the average opinion polarity of all phrases in a review.

1.1.3.2 Sentence-Level Polarity Classification

An assumption is made in much of the research on sentence-level polarity classification that one sentence expresses a single opinion from a single opinion holder. This assumption is only appropriate for simple sentences. However, for compound sentences, a single sentence may express more than one opinion (e.g., Although Fujimori was criticized by the international community, he was loved by the domestic population because people hated the corrupted ruling class). [135] uses an ***unsupervised technique*** similar to the method in [122] for polarity classification of each identified subjective sentence, but with more seed words. In [131], the problem is studied using ***supervised learning*** and considering contextual influences such as negation (e.g., “not” and “never”) and contradiction (e.g., “but” and “however”).

Other works rely on a ***sentiment word dictionaries*** which contain lists of positive and negative words that are used to match words in the opinion text. If an opinion sentence has many words from the positive dictionary, then most probably it has a positive orientation. These word lists are often used in conjunction with a set of rules or can be combined with the results of POS tagging or deep parsing. For example, in [57] a method based on WordNet is proposed. It uses few positive and negative adjectives as seeds to assign positive or negative polarities to new adjectives based on the similarity and antonymy relations defined in WordNet. A similar approach is described in [139] for movie reviews, but it uses dependency parsing to identify opinions associated with feature words. Furthermore, the polarity of a word is reversed if there is a negation relation such as "not" or "anti".

1.1.4 Joint Topic Polarity Analysis

An interesting case is when a document contains multiple topics. For instance, a review can be a comparison of two products. Or, even when a single item is discussed in a document, one can consider different features of the product to represent multiple (sub-) topics. In such a setting, it is useful to identify the opinions on each feature.

Some approaches follow a ***two-step process***. First the features that have been commented on are identified, and second the polarities are determined [58, 85, 44, 57]. These approaches attempt to identify candidate features in the opinion text with the help of POS tagging and deep parsing since features are usually noun phrases.

Other approaches ***jointly model*** topic and polarity simultaneously, or treat one as a prior for the other [39, 87, 120, 121]. These approaches mainly use probabilistic latent–factor models like PLSA [56] and LDA [22].

1.1.5 Opinion Summarization

The vast amount of opinions on the Web often overwhelm users as there is just too much information to digest. For many applications, some form of summary of the opinion mining results is needed.

A ***topic-based opinion summarization*** techniques involve generating opinion summaries around a set of topics. It consists of three steps

1. Feature identification is used to find important topics, in the text to be summarized.
2. Polarity classification is used to determine the polarities on the features found in the first step,
3. Summary generation is used to present results from the previous two steps.

One form summary, shows statistics [57, 139] that uses the results from the previous feature identification and polarity classification steps (e.g., the number of positive and negative opinions for each feature). Other studies use different granularities of summaries including words [44, 121], phrases [85] and sentences [87, 73]. Based on the discovered features using clustering and latent-topic models, the work in [85] averages the polarity prediction results of phrases for each feature as the final polarity rating for that feature. [73, 87] compute opinion trends over a timeline.

In addition, some works on summarization suggest different formats for opinion summarization different from the topic-based format [47, 68]. For example, in [68] a method is proposed that generates contrastive sentence pairs. Opinosis [47] proposes an abstractive opinion summarization method using a graph-based framework.

1.1.6 Political Opinion Mining

There is an increasing interest in political opinion mining because of their potential applications in e-Rulemaking and public opinion analysis. Analyzing political opinions on controversial topics is inherently more difficult than standard opinion mining (for products, movies, etc.), given the complexity of the topics and the subtleties in expressing opinions on them. The study described in [134] shows that the average sentiment level of USA congress debates is rather low. It is higher than that of neutral news articles, but much lower than that of movie reviews. Furthermore, affective adjectives are not the most informative indicators of political opinions. Instead, the choice of topics, as reflected in neutral nouns, is an important mode of political opinion expression by itself. The results of the study in [134] demonstrate that a significant number of political opinions are expressed in linguistically neutral tones. These characteristics make identifying the polarities an insufficient process for general-purpose political opinion classification. For example, the SentiWordNet lexicon [13], although very useful for understanding product reviews, turned out to be of little help on political texts as these have much richer phrases, rather than polarity-bearing single words like adjectives and adverbs.

There is some work on classifying political texts with regard to political parties [119, 86, 133, 38], most notably, democrats vs. republicans in the USA. For example, [86] addresses the problem of political polarity analysis as a classification problem. The classifier assigns a user's posting to a political orientation. Here, the classes typically correspond to a coarse-grained collection of bundled attitudes and beliefs (e.g., political parties). Many of these works on perspectives and viewpoints seek to extract more perspective-related information (e.g., opinion holders). The motivation is to enable multi-perspective question answering [114], where the user could ask questions such as "What is Obama's perspective on the conflict in Syria?".

There are other problems that determine whether a political speech is in support of or in opposition to the issue in a debate [19]. A related task is to classify predictive opinions in election forums into "likely to win" and "unlikely to win" [89].

Most prior research focused on classifying individual sentences into pro/con categories or on aggregating (summarizing) opinions over a large number of observations (e.g., many different people' opinions on some topic). As far to our knowledge, there is no prior work that addresses the problem of connecting opinions to individual opinion holders (typically politicians) in a systematic network of opinion holders and pro/con statements for a wide variety of fine-grained controversial topics.

1.2 Problem Description

1.2.1 Motivation

The political controversial topics being played and discussed in great depth on the news everyday such as "Greece bailout" and "Wikileaks release" have many different facets including "Ecuador grant of asylum to WikiLeaks' Assange". These topics have also many stakeholders including nations such as the "Germany", "Greece", "Ecuador", etc., or people such as "Angela Merkel", "Julian Assange", etc.. Understanding this huge "landscape" of opinions is non-trivial. There is no absolute, simple truth in answering questions such as: "Are the leaks good for democracy and transparency?", or "Would Greek default be disastrous?".

In order to understand, navigate and analyze this complicated landscape of issues and opinions taken by various stakeholders, it would be helpful if there were a browseable and queryable ***opinion-base***. Such an opinion-base would organize controversial topics according to their various, more fine-grained facets and would extract and store the opinions of people on these facets and information about who expressed an opinion on one or more topics and supported or opposed a certain direction. This kind of network could then be used to perform different kinds of analyses, such as, "Who originally supported Greece bailout, but now changed their minds?", "Which news media support Assange?", etc.. Such analyses are typically based on aggregating many statements and work well for coarse-grained topics. However, political analysts are often interested in individual and brief statements, as reported or quoted in news media, and their ***pro/con polarity*** with regard to ***fine-grained topics*** such as "deporting illegal immigrants", or "banning Facebook at workplaces". The outcome of analyzing these inputs should be a crisp set of structured records of the form: ⟨`opinion holder`⟩ ⟨`polarity`⟩ ⟨`opinion target`⟩. For example, ⟨`Angela Merkel`⟩ ⟨`support`⟩ ⟨`Greece bailout`⟩. This desired output involves aggregating statements, but only for the same pair of opinion holder and topic facet. The overall set of such records forms the opinion network.

This envisioned opinion network goes beyond the state-of-the-art opinion mining in several ways.

1. Political controversies are much more complex and opinions are often expressed in subtle forms, which makes determining pro/con polarities much more difficult than with product reviews for cameras, movies, etc. - the typical objects in prior work on opinion mining [94, 81].

2. Most prior research focused on classifying individual sentences or reviews into pro/con categories or on aggregating (summarizing) opinions over a

large number of observations (e.g., many different customers' reviews of some product). In contrast, our goal is to connect an opinion to an individual person (typically a politician) who expressed this position (multiple times, but often in very different wordings).

3. Instead of merely finding a few or the most interesting pairs of opinion holders and topics, we aim at a systematic network of opinion holders and pro/con statements for a wide variety of fine-grained controversial topics.

1.2.2 Challenges

The challenge in building the envisioned opinion network lies in the fact that the input is merely natural-language text, such as news articles or social media, where it is difficult to spot phrases that denote individual politicians or controversial topics and map them into a canonical representation. There are many forms of expressing opinions in newspapers, broadcast stations, online forums, and all kinds of social media. Acquiring political opinions from these different forms of opinions poses a set of challenges.

Challenging issues for opinion sentences: we have to acquire opinionated sentences of politicians or other stakeholders on political controversies from news articles. These opinion sentences are reported in many forms. For example "Obama supports Gay Marriage". We call this explicit form ***direct opinion***. Other form of opinions presents the beliefs or the arguments of the opinion holders which implicitly present their sentiments. For example McCain said: "The Iranians and the Russians are providing Bashar Assad with weapons. People that are being massacred deserve to have the ability to defend themselves.". We call this form **indirect opinion**. This form of opinion sentences (e.g. quotations) usually comes in short texts, so whatever linguistic features are used tend to be sparse.

Challenging issues for opinion holders: We have to identify entities expressing opinions. For example, the entity Vladimir Putin could be mentioned with different wordings like "Mr. Putin", "the Russian President", "President Putin"etc.. Therefore we need to disambiguate the different mentions.

Challenging issues for opinion polarities: we have to identify the polarity of the opinion (positive or negative, pro or con) which is not always explicitly mentioned in news.

Challenging issues for opinion targets: We have to identify the topics on which opinions are expressed. This involves the following challenges:

1. ***Political controversial nature:*** We have to acquire facets which are *politically controversial*.

2. ***Canonicalization:*** Since each facet can be expressed in different ways, we have to *canonicalize* these facets. That is, "military strikes against Iran" and "military action on Iran" are different ways of expressing the same topic and need to have a canonical form.

3. ***Fine granularity:*** Once we have canonical forms of facets we need to identify the topics to which they belong. We aim to organize politically controversial topics into a hierarchy. For example, both "sanctions against Iran" as well as "military strikes on Iran" are different *facets* of the debate on "Iran's nuclear program" which in turn can be regarded as being a part of a larger debate on "nuclear power".

4. ***Topic-dependent polarity:*** the same opinion sentence can have different polarities for different fine-grained topics, and there is a need to identify the polarity on each fine-grained topic. For example, there are two polarities in the sentence "The fence should be finished, but that mass deportations are not the answer". A positive polarity is for building the border fence, and a negative polarity is for deporting illegal immigrants.

1.3 Contributions

In this thesis, we present methods for constructing an opinion network for politically controversial topics by extracting and organizing information from online sources. Our methods address the challenges pointed out in the previous section. Our contributions can be summarized as follows.

1. **Constructing an Opinion Network:** We present methods for constructing an opinion network of controversial topics and their various facets and the stakeholders who hold opinions on these facets. We restrict our opinions to be of two types: support and oppose. Our methods for achieving this make use of a variety of building blocks and combine them in an innovative way. In particular, we integrate techniques for pattern-based information extraction with polarity classification, both customized to our setting of fine-grained political controversies. Our main method is to generate search-engine queries, and extract opinions from the snippets returned by the search engine. We conducted experimental studies with news from different media sources, which support the viability of our methods. Our methodology of constructing the envisioned opinion network was published in the ACM Conference on Information and Knowledge Management (CIKM 2011) [8], and in the ACM Conference on Web Search and Data Mining (WSDM 2012) [9].

2. **Opinion Sentences Features and Expansion Models:** We present an expressive feature model for opinions on political controversies and an expansion method of these opinions to overcome sparseness problems. We judiciously expand opinion sentences by additional words chosen from thesauri like WordNet. We also devise a feature model that distinguishes between topical terms and sentiment terms. The expansion technique treats these two groups of terms differently. The description of our features and expansion models are published in the ACM Conference on Information Retrieval (SIGIR 2010) [7], and in the ACM Conference on Information and Knowledge Management (CIKM 2012) [12].

3. **Opinion Sentences Pro and Con Classification Models:** We developed different Pro/Con classification models based on statistical language models for classifying opinion sentences into pro and con polarities for fine-grained topics. Our models also cope with the difficult case of multiple topics addressed in the same sentence and potentially different polarities for different topics. These methods are published in the ACM Conference on Information Retrieval (SIGIR 2010) [7], and in the ACM Conference on Information and Knowledge Management (CIKM 2012) [12]

4. **Opinion Network:** The output of our methods is an opinion network which has more than 100,000 opinion statements and more than 500 controversial topics organized in a Directed Acyclic Graph (DAG) with about 3,000 topic-subtopic edges. Details about this opinion network and its quality are published in the ACM Conference on Web Search and Data Mining (WSDM 2012) [9], and in the ACM Conference on Information and Knowledge Management (CIKM 2012) [12].

5. **Opinion Exploration System:** We developed a system, called "OpinioNetIt", for exploring the political opinion network and conducting more studies on it. OpinioNetIt can be used for different kinds of applications which require political opinion analysis. Here, we specifically focus on three kinds of applications: i) identifying flip-floppers who repeatedly change their minds on a topic, ii) heat map analysis indicative of political bias (news media outlets reporting stories with a certain bias), and, iii) dissenters who deviate from "expected" opinions. The description of the system and the different uses cases are published in the Workshop on Politics, Elections and Data (PLEAD) co-located with the ACM Conference on Information and Knowledge Management (CIKM 2012) [11], and in the IEEE Conference on Data Mining (ICDM 2012) [10].

1.4 Thesis Outline

We describe our opinion model in Chapter 2. We present our method for acquiring and extracting direct opinions from news in Chapter 3. Chapter 4 describes our framework for acquiring and extracting indirect opinions from news. Acquiring and organizing political controversial topics are described in Chapter 5. Our methods for generating a pro and con lexicon are described in Chapter 6. Finally, we describe our opinion exploration system in Chapter 7. We summarize our findings and point out future research directions in Chapter 8.

Chapter 2

Opinion Model

In this chapter, we explain more precisely what we mean by terms such as "opinion", "topic", "facet", etc., and formally define our goal.

2.1 Opinion

Definition 2.1 ***Opinion***
*An **opinion** is a* positive or a negative sentiment *about* a topic, or a fine-grained topic of a topic *from an* opinion holder.

2.1.1 Opinion Topic

A *topic* is informally defined as the subject matter of a particular piece of text. A *controversial topic* is a topic on which drastically varying opinions exist among people. A topic can be divided into a number of *fine-grained topics* each of which can themselves be divided further. At the most basic level, we deal with *facets* (also referred to as *raw facets*) of a topic (or fine-grained topic) and consider facets to be indivisible. Intuitively, the set of topics, fine-grained topics and facets form a hierarchy. With this understanding of the terms *topic*, *fine-grained topic* and *facet*, we now introduce some formal terminology.

Definition 2.2 ***Topic***
*A **topic** T consists of a set of **fine-grained topics** $T = \{G_1, G_2, \ldots, G_n\}$. G_i denotes a **fine-grained topic** which can be divided further into **finer-grained topics**. We denote the set of* all *topics, their fine-grained topics by $\mathcal{T} = \{T_1, T_2, \ldots, T_n\}$.*

As an example, "Nuclear Proliferation" is a topic that may have *fine-grained topics* such as "Iran's Nuclear Program", "Nuclear Non-Proliferation Treaty", etc..

Definition 2.3 ***Facet***
*A **facet** f_i denoted by a string $\mathtt{str}(f_i)$ is a **finest-grained topic**, or a **fine-grained topic** at the most basic level which is considered to be indivisible.*

For example, "Iran's Nuclear Program" is a *fine-grained-topic* which in turn has *facets* such as "military strikes on Iran" and "sanctions on Iran".

Definition 2.4 ***Canonicalized Facet***
*A **canonicalized facet** F_i denoted by a string $\mathtt{str}(F_i)$ is an equivalence class of facets $S(F_i) = \{f_1, f_2, \ldots, f_k\}$ where each f_j has the same semantic meaning. It is a **fine-grained topic** G of a topic T.*

For example, "military strikes on Iran" and "military action against Iran" are both semantically the same and we may choose the former as the canonical form.

Definition 2.5 ***Topic Hierarchy***
*A **topic hierarchy** is a DAG consisting of* nested *topics, fine-grained topics and facets. Formally, let $TH = \{V, E\}$, where $V = \{v_i | v_i \in \mathcal{T}\}$ and $E = \{(v_i, v_j) | v_j$ is a fine-grained topic or facet of $v_i\}$.*

As an example, "Nuclear Technology" is a topic that may have *finer-grained* topics such as "Nuclear Proliferation", "Nuclear Energy", "Radiation Effects", "Nuclear Proliferation" in turn may have more *finer-grained* topics such as "Iran's Nuclear Program", "Nuclear Non-Proliferation Treaty", etc.. And "Iran's Nuclear Program" in turn has *facets* such as "military strikes on Iran" and "sanctions on Iran".

2.1.1.1 Opinion Target

Definition 2.6 ***Opinion Target***
*An **opinion target** O is a topic T, a fine-grained topic G of a topic, or a facet F of a topic, on which the opinion is expressed.*

For example, in the sentence "Obama supports the military actions in Libya", the *opinion target* is "military actions in Libya".

2.1.2 Opinion Polarity

Definition 2.7 ***Opinion Polarity***
An ***opinion polarity*** P *on a topic* T*, a fine-grained topic* G*, or a facet* F *is either a* ***positive*** *or a negative sentiment.*

Definition 2.8 Pro ***Opinion***
A pro ***(support)*** *opinion on a topic* T*, a fine-grained topic* G*, or a facet* F *is an opinion with* ***positive*** *polarity.*

Definition 2.9 Con ***Opinion***
A con ***(oppose)*** *opinion on a topic* T*, a fine-grained topic* G*, or a facet* F *is an opinion with* ***negative*** *polarity.*

2.1.3 Opinion Holder

In the context of opinion mining and sentiment analysis, the opinion holder is also called opinion source.

Definition 2.10 ***Opinion Holder***
An ***opinion holder*** H *is an entity such as a person, organization or government that holds an opinion on for on a topic* T*, a fine-grained topic* G*, or a facet* F*. We denote the set of all opinion holders as* $\mathcal{H} = \{\mathsf{H}_1, \mathsf{H}_2, \ldots, \mathsf{H}_n\}$.

For example, in the sentence “Obama supports the military actions in Libya”, the *opinion holder* is ‘Obama”.

2.1.4 Opinion Triple

Definition 2.11 ***Opinion Triple***
An ***opinion triple*** *is denoted by:*
$\langle \mathsf{H} \rangle \ \langle \mathsf{P} \rangle \ \langle \mathsf{O} \rangle$
where H *is the opinion holder holding an opinion polarity* P *on an opinion target* O.

For example, ⟨`Obama`⟩ ⟨`support`⟩ ⟨`Gay Marriage`⟩ is a triple derived from the sentence “Obama supports Gay Marriage”.

2.1.5 Opinion Quadruple

Definition 2.12 ***Opinion Quadruple***
*An **opinion quadruple** is denoted by:*

$\langle \mathsf{H} \rangle\ \langle \mathsf{P} \rangle\ \langle \mathsf{O} \rangle \langle \text{context} \rangle$

where the triple: $\langle \mathsf{H} \rangle\ \langle \mathsf{P} \rangle\ \langle \mathsf{O} \rangle$ *is extracted from the textual* context.

For example, $\langle \texttt{Obama} \rangle\ \langle \texttt{support} \rangle\ \langle \texttt{Gay Marriage} \rangle \langle \textit{Obama supports Gay Marriage} \rangle$ is a quadruple derived from the sentence "Obama supports Gay Marriage".

2.1.6 Opinion Network

Definition 2.1.1 *An* **opinion network** *is a node- and edge-labeled, directed graph* $\mathsf{ON} = (\mathsf{V}, \mathsf{E}, \mathsf{C})$. *A node represents an opinion holder, a topic, a fine grained topic, or a facet, that is* V $\mathsf{V} = \{v_i | v_i \in \mathcal{T} \cup \mathcal{H}\}$, *where* $\mathcal{T}$ *is the set of topics,* $\mathcal{H}$ *is the set of opinion holders. An edge* $\mathsf{E} = \{(v_i, v_j, l) | v_i \in \mathcal{H} \cup \mathcal{T}, v_j \in \mathcal{T}, l \in \{\mathsf{support}, \mathsf{oppose}, \epsilon\}\}$, *where* l *is the edge label.* $\mathsf{C} : \mathsf{E} \to \{\mathsf{str}_1, \mathsf{str}_2, ..., \mathsf{str}_n\}$ *maps each edge to a context.*

The topic hierarchy TH *is part of the opinion network (therefore, we have edges with both endpoints in* $\mathcal{T}$ *and with empty labels). Note that the* opinion triples *each correspond to a labeled edge in the graph and the mapping* C *associates each triple to its context.*

An example of what the opinion network looks like in the context of the recent "Arab Spring" is shown in Figure 5.2. The different boxes represent nodes for facets, fine-grained topics, and topics. Facets such as "U.N. Mandate against Libya", "use of force against civilians in Libya", etc., have fine-grained topics "Military Intervention in Libya" and "Gaddaffi's Response to the Civil War" respectively. The fine-grained topic in turn are part of larger topics "Libyan Civil War", "Syrian Uprising", etc.. At the top level of topics, we have the "Arab Spring". At the other end of the network, there are the opinion holders nodes such as "Barack Obama", "Hilary Clinton", etc., which are connected to the facets through labeled edges. The labels of the edges represent the opinions polarities.

2.2 Opinion Sentences

Opinion holders, polarities, and opinion targets are extracted from their opinion contexts. The context is an opinionated sentence or paragraph reported in news articles. Politicians or other stakeholders opinions on political controversies in news articles are reported in many forms. We are interested in two main forms: the direct form and the indirect form.

2.2.1 Direct Opinions

The direct form of opinions is derived from a sentence that has three distinct components: i) the opinion holder, ii) the opinion polarity, and iii) the opinion target on which the opinion is expressed. These forms of opinions precisely mention wether an opinion holder is for or against an opinion target (e.g. "Obama supports Gay Marriage").

Definition 2.13 ***Direct Opinion***

*A **direct opinion** is a* **quadruple** *derived from a sentence which has, i) one opinion holder* $\mathsf{H} \in \mathcal{H}$ *or more ii) one topic or more* $\mathsf{T} \in \mathcal{T}$ *or any of its fine-grained topics* G*, and facets* F*, iii) an explicit opinion polarity* P *(pro or con) on each topic* T*, fine-grained topic* G*, and facet* F*.*

Table 2.1 lists some examples from news articles for sentences for deriving direct opinions.

The United Nations Security Council on Sunday unanimously condemned ***the Syrian government for its role in the massacre of at least 108 villagers***
Russia has typically rejected ***any international effort to support the opposition in a way that might repeat the NATO military intervention in Libya***
Turkish Prime Minister Recep Tayyip Erdogan has censured ***the government of Syrian President Bashar al-Assad over its approach to reforms in the Arab country***
Obama condemns ***al-Assad's 'disdain for human life and dignity'***
Russia sides firmly with ***Assad***

Table 2.1: Subjective sentences in news articles for deriving direct opinions

2.2.2 Indirect Opinions

On the other hand, a sentence which is used to derive indirect form of opinions has two distinct components: i) an opinion holder, and ii) an ***opinion argument***. An ***opinion argument*** is a type of linguistic subjectivity, where an opinion holder is arguing for or against something or expressing a belief about what is true, should be true or should be done in his or her view of the world [131, 111]. The opinion target, and the opinion polarity of the opinion holder on the target is not necessarily and precisely mentioned, but is to be identified from the argument.

Definition 2.14 ***Indirect Opinion***
An ***indirect opinion*** *is a* **quadruple** *derived from a* **paragraph or a sentence**. *Each* **paragraph or sentence** *has i) an* **opinion holder** $H \in \mathcal{H}$, *and ii) an* **opinion argument** *by the opinion holder. Each* **opinion argument** *addresses one or more topic* T, *fine-grained topic* G, *or facet* F. *The opinion polarities (pro/con) on the topics in the* **opinion argument** *are not necessarily mentioned explicitly, but are derived from the argument.*

Table 2.2 lists some examples from news articles for opinions which fall under the indirect form of opinions.

Carolyn Maloney said “Guns kill. And those who glamorize gunplay or worship gun ownership do no service to humanity.”
“homosexuality poses a serious threat to family. The bill has helped raise public awareness about the dangers to our children” **said David Bahati**
“The death penalty provides a sense of justice to the system, is a just punishment for murder and has a deterrent effect on crime” **said Cassell**
Quan also said “The Bay Area Occupy Movement has got to stop using Oakland as their playground”
“Shining a light on 45 years of US ’diplomacy,’ it is time to open the archives forever” **By WikiLeaks**

Table 2.2: Indirect opinions in news articles

Chapter 3

Direct Opinion Extraction

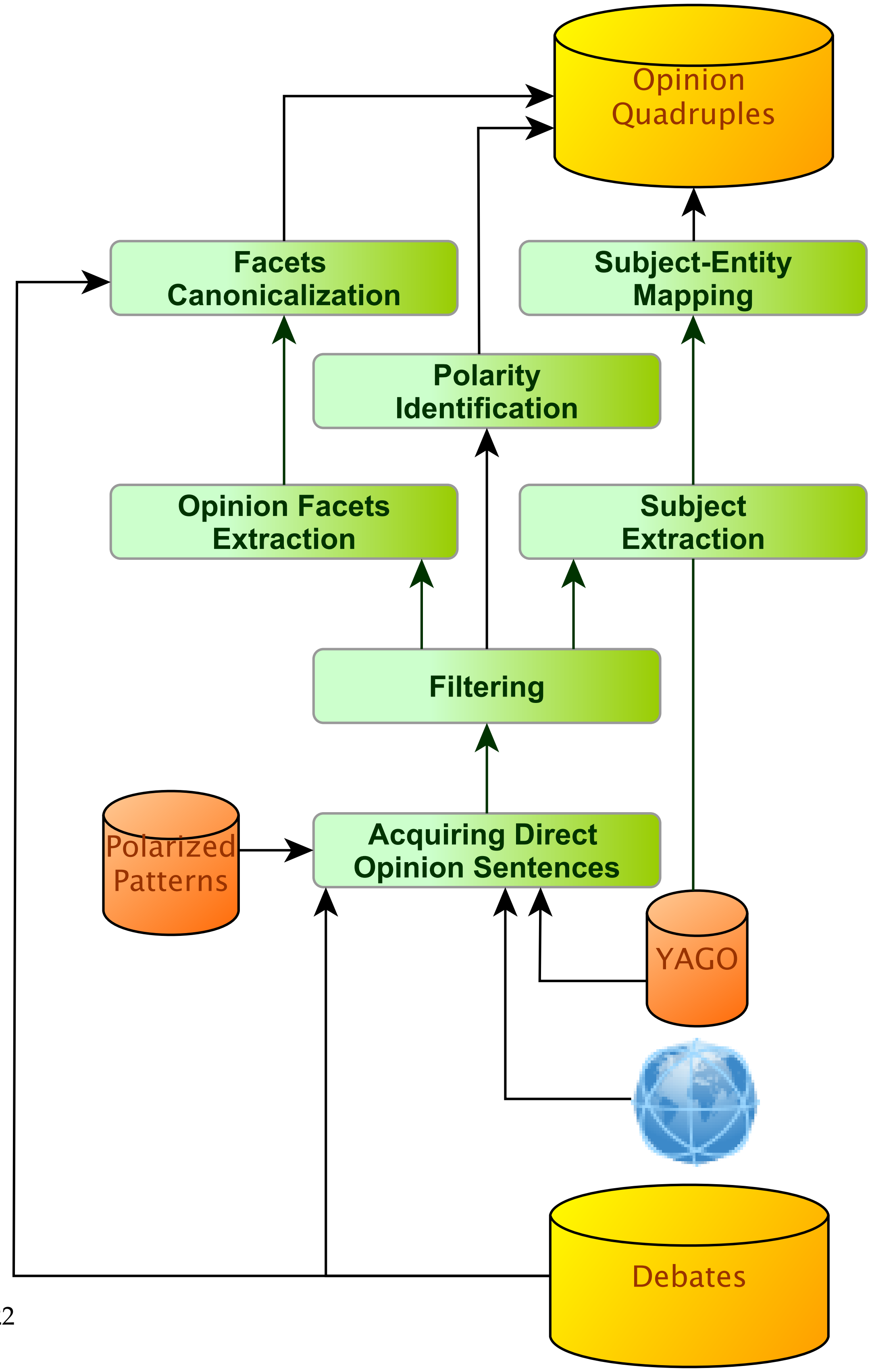
3.1 Introduction

Politicians or other stakeholders opinions on political controversies are reported on the web in many forms. As it was mentioned in Chapter 1, we are interested in two main forms of opinions: the direct form and the indirect form. In Chapter 2, we formally defined and described these two types of opinions. In this chapter, we describe our approach to acquire direct opinions from news articles.

In the rest of this Chapter, we first describe our approach to acquire direct opinion sentences in Section 3.2. In Section 3.3, details of the extraction process for opinion holders, opinion polarities, and opinion targets are given. Section 3.4 presents our experimental evaluation. Section 3.5 positions our contributions with regard to related work. We conclude with Section 3.6.

3.2 Acquiring Opinions from the Web

Our aim is to acquire a large number of politicians or other stakeholders opinions on a wide variety of fine-grained politically controversial topics. While using Wikipedia is an option from which we can acquire opinions, we believe that comprehensive coverage of topics is possible only on the Web. Therefore, we decide to use the Web to gather a large number of opinions, while Wikipedia is used as a resource to help organize the topics as we will describe in Chapter 5. Our initial approach to acquiring opinions uses the whole Web. However, this initial approach returns noisy opinion sentences which can not always be parsed correctly (e.g., user comments in forums). Therefore, our pragmatic approach is


Opinion Quadruples
Facets Canonicalization
Subject-Entity Mapping
Polarity Identification
Opinion Facets Extraction
Subject Extraction
Filtering
Polarized Patterns
Acquiring Direct Opinion Sentences
YAGO
Debates

to still query the Web, but to limit the sites from which results are obtained. Figure 3.1 gives an overview of our methodology. Our current resources are mainly online news media.

3.2.1 Direct Opinion Sentences

We describe in this section our approach to acquiring opinions of the direct form. Before that, we explain the structure of a direct opinion sentence.

3.2.1.1 The Structure of Direct Opinion Sentence

Examples of direct opinions in online news are listed in Table 3.1. The opinions listed in the table have three main components:

1. The subject: which is the opinion holder (e.g. Obama, Sarkozy, etc.),
2. The polarized verb phrase: which is the opinion polarity (e.g. supports, opposes, condemns, etc.),
3. The object: which is the opinion target (e.g. lowering of abortion limit, Assad departure, etc.).

Definition 3.1 ***The Subject-Polarity-Object Structure***
*The **Subject-Polarity-Object** structure of an opinion sentence denoted as ⟨**SPO**⟩ consists of a (1) ⟨**S**ubject⟩ which is the opinion holder, (2) a ⟨**P**olarized verb phrase⟩ which is the opinion polarity, and (3) an ⟨**O**bject⟩ which is the opinion target.*

3.2.1.2 Acquiring Direct Opinion Sentences

The basic structure of the direct opinion sentences gives us a hint that we can collect similar opinion sentences by looking for sentences in news resources that follow the same structure. All regular English sentences have subjects, verb phrases, and objects. What make opinion sentences different are their *polarized verb phrases and therefore, they are our main targets in news resources.*

Our initial approach to acquiring opinion sentences uses only few polarized patterns which are listed in Table 3.2. These patterns are submitted as phrase queries to search engines (e.g. Google and Bing) in three different ways.

1. **The United Nations Security Council** on Sunday unanimously condemned *the Syrian government for its role in the massacre of at least 108 villagers*
2. **Russia** has typically rejected *any international effort to support the opposition in a way that might repeat the NATO military intervention in Libya*
3. **Turkish Prime Minister Recep Tayyip Erdogan** has censured *the government of Syrian President Bashar al-Assad over its approach to reforms in the Arab country*
4. **Obama** condemns *al-Assad's 'disdain for human life and dignity'*
5. **Obama** opposes *ban on sex-selective abortions*
6. **Mr. Obama** opposes *banning abortion based on gender*
7. **President Obama** opposes *N.C. marriage amendment*
8. **Obama** voted for a *$100 million education initiative to reduce teen pregnancy and provide contraceptives to young people*
9. **GOP Intel Committee Chair** is against *arming rebels in Syria*
10. **McCain** calls for *US to support Syria rebels*
11. **McCain** criticizes *Obama administration Iran policy*
12. **Russia** is not against *Assad departure*
13. **French President Nicolas Sarkozy** has spoken out strongly against *the wearing of the burka by Muslim women in France*
14. **Merkel** supports *stronger European investment bank*
15. **Merkel** supports *NATO's total pullout from Afghanistan by 2014*
16. **Merkel** supports *eurozone 'red card'*
17. **Cameron** says he's not opposed to *U.K. referendum on EU*
18. **Cameron** supports *Sarkozy as president*
19. **David Cameron** supports *gay marriage legalization*
20. **Prime Minister David Cameron** supports *lowering of abortion limit*
21. **Hollande** supports *EU Banking Union*
22. **French President Hollande** supports *UN intervention in Mali*
23. **Hollande** criticizes *Greek Austerity Plan*
24. **France's Hollande** opposes *strategic oil release*
25. **Peres** supports *'provisional' Palestinian state*
26. **U.S.** supports *plan to suspend sanctions if Iraq disarms*

Table 3.1: Direct opinion sentences in online news

1. *A polarized pattern query: the pattern is submitted as a phrase query. For example "voted for", and "voted against".*

2. *A named entity and a polarized pattern query: for a predefined list of politician names obtained from the Yago ontology [116], we formulate one phrase query for each politician last name and each pattern. For example "McCain voted against", and "Obama supports".*

3. *A polarized pattern and a topic query: for a predefined list of political controversial topics collected from the web, we formulate one phrase query for each pattern and each topic. For example "opposes same-sex marriage", and "supports abortion".*

All the returned results snippets along with their titles and their news articles are collected.

"supports", "supported", "voted for", "voted against", "opposes", "opposed"

Table 3.2: Initial set of polarized patterns for the opinion extraction process

3.2.2 Filtering Opinion Sentences

The collected results snippets represent the initial set of candidate opinion sentences. Some of these sentences lack the semantics and the structure we target in such opinion sentences. Therefore a filtering process is required. We filter these sentences by both semantic and structure (see Algorithm 3.1 describing the filtering process).

3.2.2.1 Filtering by Semantics

Search engines return relevant snippets to a query and highlight the matched query keywords using a boldface font. If all *the query keywords are highlighted in the snippet, then the snippet is added to the candidate list, otherwise we filter it out. For example, for the phrase query "voted against", Google returns a list of result snippets as shown in Table 3.3. We ignore the second snippet since the query keywords are not highlighted.*

3.2.2.2 Filtering by Structure

We are interested in collecting direct opinion sentences that follow the $\langle \boldsymbol{SPO} \rangle$ *structure. Some of the collected snippets contain the queries' keywords, but still do not follow this structure. Therefore we further filter out these snippets. In order to do this, the*

Algorithm 3.1 Filtering candidate opinion sentences

Input: $S \Leftarrow \{s_1, s_2, \dots, s_n\}$ ▷ the snippets of the search results
Input: $Q \Leftarrow$ "$q_1 q_2 \dots q_k$" ▷ the query
Output: C ▷ the set of candidate snippets

$I \Leftarrow \emptyset$ ▷ the initial set of candidate snippets
for $s_i \in S$ **do**
  if $(s_i.\mathtt{indexOf}(Q) \geq 0)$ **then**
    $I \leftarrow I + s_i$
  end if
end for
$C \Leftarrow \emptyset$
for $s_i \in I$ **do**
  $s_i = s_i.\mathtt{replace}(Q, \text{"supports"})$
  $D_i \leftarrow \mathtt{parse}(s_i)$ ▷ the set of typed dependencies of s_i
  if ($\mathtt{nsubj}_i$("supports",subject) $\in D_i$ & $\mathtt{dobj}_i$("supports",object) $\in D_i$) **then**
    $C \leftarrow C + s_i$
  end if
end for
return C

individual sentences of the resulting snippets are parsed using the Stanford parser [71]. It is a high-accuracy statistical phrase structure parser trained on the Penn Wall Street Journal Treebank. The Stanford parser is employed for the generation of the parse tree and the typed dependencies. Before we parse the sentences as a preprocessing step, long sentences with more than one polarized pattern are broken up into shorter sentences, each with a single polarized pattern. For example, the sentence, "Clinton opposes Iran nukes, and supports Palestinian government" is divided into two sentences: "Clinton opposes Iran nukes" and "Clinton supports Palestinian government". Next, each occurrence of a polarized pattern (e.g. voted for, supported, opposes) in the snippet is replaced by the polarized pattern "supports". We do this replacement in order to increase the parsing precision. Table 3.4 shows an example of the output of the syntactic analysis of the Stanford parser.

The parsing dependencies provide a representation of grammatical relations between words in a sentence. The dependencies are triplets: name of the relation, governor and

Snippet (1) Asian markets fall on U.S. bailout failure - CNN.com edition.cnn.com/2008/BUSINESS/09/29/us.congress.../index.htmlCached - Similar *29 Sep 2008 – U.S. lawmakers in the House of Representatives on Monday **voted against** the biggest proposed government intervention in the U.S. economy ...*
Snippet (2) Explainer: What next for the bailout plan? - CNN.com edition.cnn.com/2008/BUSINESS/09/30/us.bailout.../index.htmlCached - Similar *30 Sep 2008 – The U.S. Senate voted Wednesday by 74 votes to 25 to approve a $700 billion economic bailout plan. The proposals would allow Treasury ...*
Snippet (3) Japanese lawmakers approve sales tax increase amid political - CNN edition.cnn.com/2012/06/26/world/asia/japan-tax.../index.html?... *26 Jun 2012 – But 57 members of Noda's own party, the Democratic Party of Japan, **voted against** the sales tax increase, according to the office of the lower ...*
Snippet (4) Craig: I did nothing 'inappropriate' in airport bathroom - CNN.com edition.cnn.com/2007/POLITICS/08/28/craig.arrest/Cached - Similar *28 Aug 2007 – Craig also has opposed expanding the federal hate crimes law to cover offenses motivated by anti-gay bias and, in 1996, **voted against** a bill ...*
Snippet (5) Romney jabs at Santorum's record in CNN debate - CNN.com edition.cnn.com/2012/02/22/election/2012/arizona.../index.html *23 Feb 2012 – Santorum '**voted against** his principles'. Analysts: Romney outperformed Santorum. Santorum, Romney clash on earmarks. Birth control ...*

Table 3.3: Examples for snippets returned by Google for the phrase query "voted against" with domain restricted to CNN news

*dependent. For example in Table 3.4, and for the triplet "nsubj(supports-2, Obama-1)" the grammatical relation is "subject", the governor is "supports", and the dependent is "Obama. We rely on the typed dependencies in order to filter out sentences that do not follow the ⟨**SPO**⟩ structure. We assume that a sentence that follows the ⟨**SPO**⟩ structure has among its typed dependencies two triplets of the form **nsubj(supports, subject)**, and **dobj(supports, object)**. Otherwise, we assume that the sentence violates the structure and should be filtered out. The underlined triplets in Table 3.4 indicate that*

Sentence: **The U.S. President Barack Obama** opposes ***ban on same-sex marriage***
Sentence with replaced polarized pattern: **The U.S. President Barack Obama** supports ***ban on same-sex marriage***

Parse Tree:
 	
(ROOT
  (S
    (NP (DT The) (NNP U.S.) (NNP President) (NNP Barack) (NNP Obama))
    (VP (VBZ supports)
      (NP
        (NP (NN ban))
        (PP (IN on)
          (NP (JJ same–sex) (NN marriage))))))))

Typed Dependencies: **det**(Obama-5, The-1) **nn**(Obama-5, U.S.-2) **nn**(Obama-5, President-3) **nn**(Obama-5, Barack-4) **nsubj**(supports-6, **Obama**-5) **root**(ROOT-0, supports-6) **dobj**(supports-6, ***ban***-7) **prep**(ban-7, on-8) **amod**(marriage-10, same-sex-9) **pobj**(on-8, marriage-10)

Table 3.4: Stanford syntactic analysis output

*the sentence in the example follows the ⟨**SPO**⟩ structure. Table 3.5 shows a sentence that violates the structure, where we notice that its typed dependencies lack the two required grammatical relations that guarantee the ⟨**SPO**⟩ structure.*

At this point we do not claim that all the remaining candidates are actually good ones for one main reason. Some of the candidates follow the structure we target, but their topics are not of political nature. Our goal is to collect opinions on politically controversial topics for politicians and other stakeholders. Some of our initial patterns such

Sentence: Online <u>support</u> groups and forums also are available, where you can talk with people
Sentence with replaced polarized pattern: Online <u>supports</u> groups and forums also are available, where you can talk with people
Typed Dependencies: **nsubj**(<u>supports</u>-2, **Online**-1) **root**(ROOT-0, supports-2) **nsubj**(available-8, groups-3) **cc**(groups-3, and-4) **conj**(groups-3, forums-5) **advmod**(available-8, also-6) **cop**(available-8, are-7) **ccomp**(supports-2, available-8) **advmod**(talk-13, where-10) **nsubj**(talk-13, you-11) **aux**(talk-13, can-12) **advcl**(available-8, talk-13) **prep**(talk-13, with-14) **pobj**(with-14, people-15)

Table 3.5: Stanford typed dependencies for a sentence violating ⟨SPO⟩ structure

as "supports" could return results snippets like "Canon's PowerShot camera supports 10X zoom", "Acer notebook supports security cards, wireless", and "Preview of Liquid System 5.0 supports MP3". We will describe in Chapter 5 our solution to filter out these examples.

3.3 Opinion Quadruple

A candidate opinion quadruple as it was defined in Chapter 2, is formed from a candidate opinion sentence by adding the triple of the opinion holder, the opinion polarity, and the opinion target. In addition, the snippet used to extract the opinion triple is added as its context. In this section, we describe how we identify the opinion holder, the opinion polarity, and the opinion target of a candidate opinion sentence using its parse tree and its typed dependencies obtained as an output of the filtering process described in Section

3.2.2.

3.3.1 Identifying Opinion Holder

In order to identify the opinion holders, we first identify the subject of the opinion sentence, and then map the subject to an entity (see Algorithm 3.2 for an overview).

To identify the subject, we refer to the typed dependency relation "subj(supports, subject)" of an opinion sentence. The subject of the sentence, can be a pronoun such as "he", "she", "I", "we", "they", etc., or a noun such as "president", "Obama", etc.. We treat each case differently.

Algorithm 3.2 Identifying Opinion Holder

Input: s_i ▷ a snippet $\in C$ the set of candidate snippets
Input: D_i ▷ the typed dependencies set of a snippet s_i
Input: T_i ▷ the parse tree of a snippet s_i
Input: $\mathit{subject}_i$ ▷ subject extracted from nsubj_i("supports",subject) $\in D_i$ of s_i
Input: Yago ▷ A knowledge-base
Output: H_i ▷ the opinion holder of s_i

```
pronouns ⇐ {"he", "she", "we", "they", etc.}
if subject_i ∈ pronouns then
    if (nsubj_i(verb,non-pronoun-subject_i) ∈ D_i & verb ≠ "supports") then
        subject_i ⇐ non-pronoun-subject_i
    else
        discard s_i
        return H_i ⇐ ∅
    end if
end if
subject_i ⇐ the noun-phrase of subject_i extracted from T_i
if subject_i ∈ means triples of Yago then
    return H_i ⇐ h_i : h_i with Max(weight(means(subject_i, h_i)))
else
    discard s_i
    return H_i ⇐ ∅
end if
```

```
(NP
    (DT The)
    (NNP U.S.)
    (NNP President)
    (NNP Barack)
    (NNP Obama)
).
```

3.3.1.1 Noun Subject

*The noun word that appears in the subject relation of the typed dependency of an opinion sentence can be part of a noun phrase. In this case, the subject of the opinion is represented by this noun phrase rather than the single word in the typed dependency relation. For example the sentence in Table 3.4, has the typed dependency relation "nsubj(supports-6, Obama-5)", which indicates that the subject is "Obama". But in the original sentence, we notice that the noun phrase "The U.S. President Barack Obama" is the actual subject. Therefore using the parsed tree of the sentence, we target the top parent noun phrase that includes the single word in the subject relation of the typed dependency. In the parse tree, a noun phrase is denoted as "**NP**". For example, the parse tree of the sentence in Table 3.4, indicates that the top parent noun phrase of "Obama" is:*

3.3.1.2 Pronoun Subject

*For opinion sentences with pronoun subjects such as "he", "she", "I", "we", "they", etc., in their typed dependency relation "subj(supports, subject)", it is often the case that the same sentence also names the opinion holder. For example, "Shimon Peres says he opposes military strikes on Iran" contains both the polarized pattern "opposes" as well as the opinion holder "Shimon Peres". On the other hand, the sentence can have many named entities, for example "President Obama also told Roberts that he supports gay marriage", contains two named entities. In such sentences with one or more named entities we rely on a heuristic which refers to the typed dependency of the sentence with the relation "subj" having a non-pronoun dependent. For example, the non-pronun subject relations of the two sentences given above are "**nsubj(says-3, Peres-2)**", and "**nsubj(told-4, Obama-2)**" respectively. Therefore Peres, and Obama, are the subjects.*

Again, we consider the top noun phrases including the nominated subjects as we described in the previous paragraph. If no such subject relation exists among the typed dependencies of the sentence, we discard the sentence. Such sentences require a larger context in order to identify the subject using co-reference resolution algorithms which we do not consider in this research.

3.3.1.3 Subject-to-Entity Mapping

In order to ensure that the identified subject noun phrase is indeed a named entity, we use the Yago ontology [116] which comprises all individual entities in Wikipedia and additionally provides a `means` *relation which maps variations of names to the correct entities. For example, "B. Obama" and "President Obama" would both map to the entity* `Barack_Obama` *through the* `means` *relation. The* `means` *relation itself is constructed from Wikipedia cues like redirects and href anchor texts [55].*

As many names are ambiguous, Yago actually connects the surface names to all possible *meanings. For example, for "Obama" it provides both* `Barack_Obama` *as well as* `Michelle_Obama` *as entity candidates. Fortunately, Yago comes with a simple but powerful heuristics for the* preferred meaning *of a name: the entity which most frequently occurs in Wikipedia as a link target for an href anchor text with the given name.*

Using Yago to identify names has the added advantage of canonicalizing the names of the opinion holders. In the case that Yago does not know a name at all, then we discard the snippet. The disambiguation heuristics mentioned above may look crude, but it works extremely well for important stakeholders like politicians or organizations. More advanced methods for named entity disambiguation can be easily plugged into our architecture.

3.3.2 Opinion Polarity

*As we described in Section 3.2.1, we rely on an initial set of polarized patterns to acquire opinion sentences. Therefore, the opinion polarity (e.g. support or oppose) is determined by the polarized pattern in the sentence. But we additionally check wether the present polarized pattern is actually negated in the sentence. In such a case, we use the opposite polarity (e.g. "doesn't support" means "oppose"). Negated polarized patterns in a sentence can be identified by the "**neg(polarized verb, negation word)**" typed dependency with two arguments, the polarized verb, and the negation word used to negate the* *polarized verb (e.g. not, never, etc.). For example, the sentence "Barack Obama doesn't supports Marijuana", has among its typed dependencies the relation "**neg(support-5, n't-4)**".*

The number of opinion quadruple that we collect is limited by the initial polarized pattern set, since only those patterns are queried. For example, we find "supports", but we would miss "in favor of". In Chapter 6, we present several methods that we developed to automatically identify a large number of support and oppose patterns and create a lexicon of polarized phrases that can be used to collect more opinions from the Web.

3.3.3 Identifying Opinion Target

In order to identify the opinion target, we first identify the object representing the facet of the opinion sentence, and then canonicalize this facet. Table 3.6 gives examples of facets and their canonical forms. In this section, we will describe the approach we follow to identify the facets of the opinion sentences. In Chapter 5. we will describe the canonicalization process of the facets.

*To identify the object representing the opinion facet, we refer to the typed dependency relation "**dobj(supports, object)**" of an opinion sentence. The noun word that appears in the object relation of the typed dependency of an opinion sentence can be part of a noun phrase. In this case, the object of the opinion is represented by this noun phrase rather than the single word in the typed dependency relation. For example the sentence in Table 3.4, has the typed dependency relation "**dobj(supports-6, ban-7)**", which indicates that the object is "ban". But in the original sentence, we notice that the noun phrase "ban on same-sex marriage" is the actual object. Therefore using the parsed tree of the sentence, we target the top parent noun phrase that includes the single word in the object relation of the typed dependency. For example, the parse tree of the sentence in Table 3.4, indicates that the top parent noun phrase of the word "ban" is:*

3.4 Experimental Evaluation

For evaluating the accuracy of our automated methods, we present experimental studies of the output quality of extracted opinions.

```
(NP
    (NP
        (NN ban)
    )
        (PP
            (IN on)
            (NP
                (JJ same-sex)
                (NN marriage)
            )
        )
)
```

Facet	**Fine-grained Topic**
a military strike by America or Israel on Iran	***Military actions against Iran***
the peaceful nuclear program of the Islamic Republic	***Nuclear program of Iran***
a military strike on Iran	***Military action against Iran***
put sanctions on Iran	***Sanctions against Iran***
No-Fly Zone over Libya	***Military action in Libya***
using military force against the regime of Libyan leader	***Military action in Libya***
use of force against civilians in Syria	***Massacres in Syria***
arming the Syrian opposition	***Arming rebels in Syria***
Russian Arming of Assad	***Arming Assad regime***
arming Syrian president Bashar Assad's military	***Arming Assad regime***

Table 3.6: Examples for facets and their fine-grained topics (canonicalization)

3.4.1 Data

*For the experiments conducted in this section, a lexicon of support/oppose phrases, shown in Table 3.10, served as our polarized patterns to extract ca. 30,000 opinion statements. Our online news sources are Aljazeera (**ALJ**), BBC, CNN, and New York Times (**NY**). In addition we used the same patterns to collect opinions from Wikipedia (**WP**), and Google News (**GN**). Table 3.8 shows the distribution of the opinions. We restricted the number of snippets collected to the first 1,000 search engine results for* *each pattern. These snippets yielded ca. 14,000 named entities, 23,000 facets, and 4,000 canonicalized facets.*

3.4.2 Methodology

We performed the following experiments for which we present precision results:

1. *A* random sample *of 2,005 opinion statements (opinion target-opinion holder pairs). A total of 18 human judges assessed these opinions as "correct" or "incorrect".*

2. *A small* focused sample *of 50 opinion statements by prominent politicians on important contemporary topics.*

3. *We compared our opinion network to a* limited ground-truth set *on prominent US politicians and their opinions about major topics, available on* `procon.org` *and* `ontheissues.org`*.*

3.4.3 Metrics

The main measure of interest is the precision of the opinion statements, the fraction of truly correct outputs that our methods yield. We estimate precision by samples, with Wilson confidence intervals for statistical significance [130]. Let TP *denotes the number opinions that were judged as* correct*, while* FP *is the number of opinions that were judged as* incorrect*. The* precision *is then computed as:*

$$Precision = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}} \tag{3.1}$$

Let FN *be the number of missed opinions. Then, the* recall *should be computed as:*

$$Recall = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}} \tag{3.2}$$

But, it is not obvious how to estimate FN *here, without manually reading the entire corpus. Thus we restrict this aspect of our studies to giving absolute numbers of different outputs obtained by our methods.*

3.4.4 Evaluation Method for Random Sample

3.4.4.1 Setup

As mentioned before, our metric of interest is the accuracy (or precision) of the opinions. Since it was not possible to evaluate all extracted opinions, but only a sample, we generalized the precision of the sample to the precision of the all collected opinions with the Wilson confidence interval [130].

We set up a website where human judges could log in and we presented each judge with a randomly selected opinions, for which they have to assess the correctness. Since the judges might not have enough knowledge to assess each opinion, the search result snippet and the page from which the opinion was extracted were presented next to the opinion. Thus, the judges evaluated the correctness of opinion with respect to the content of the above described resources. We did not assess the factual correctness of the resources itself.

From our own inspection, it was clear that there were several opinions which could not be easily evaluated as "correct" or "incorrect". Therefore, we devised alternative judgements as follows. For opinions that the judges deemed correct, they could additionally indicate the quality of the named entity and the facet. Specifically, they could state that:

1. *The named entity was not fully/correctly extracted,*
2. *The facet was not fully/correctly extracted,*
3. *The facet is too generic.*

The last option was to indicate facets such as "the bill" (as in, `X supports the bill`*) which may be correct, but does not deliver any meaningful information. For opinions that the judges deemed incorrect, they could indicate whether:*

1. *The named entity is incorrect,*
2. *The facet is incorrect,*
3. *The relation is incorrect.*

From each source, we drew a random sample of opinions and add it to one overall sample. We look for all opinions from the different sources as if they are in one pool and therefore the overall sample is a random sample of the pool. This allowed us to *estimate the overall correctness of the opinions in the pool. The sample of the source may contain opinions extracted by different relations (support/oppose) using different polarized patterns. Since samples were randomly drawn, we expect the distribution of the relations and the extraction patterns in the samples to represent the distribution of relations and patterns in the pool.*

3.4.4.2 Results

Over the course of a week, 18 judges evaluated an overall number of 2,005 opinions. Each opinion was evaluated by exactly two judges. This gave us a total precision value of the overall sample. We generalize the precision on the sample to the precision of the pool by help of the Wilson confidence interval, and get a center of about 0.724% at an interval width of less than $\pm 0.02\%$*. This ensures that our findings are statistically significant. The inter-annotator agreement results (the Cohen's kappa coefficient) are shown in Table 3.7. Table 3.8 shows the precision with respect to the different sources, and with respect to the type of relation (support/oppose). Table 3.10 shows the precision with respect to the different polarized patterns. Table 3.9 shows other details. For the entire set of 2,005 samples, the micro-averaged precision is 72.4%. Given the difficulty of this extraction task, these results are quite satisfying.*

		B	**B**
		Correct	Incorrect
A	**Yes**	1448	82
A	**No**	0	475
kappa coefficient=			**0.89**

Table 3.7: Inter-annotator agreement of random sample

Source	**Overall**	**Support**	**Oppose**	**ALJ**	**BBC**	**CNN**	**GN**	**NY**	**WP**
#Opinions	29648	16011	13637	970	3349	2650	6861	9877	5941
#Evaluated	2005	1121	884	364	349	308	301	364	319
Precision	0.724	0.70	0.75	0.72	0.66	0.78	0.788	0.74	0.69

Table 3.8: Opinions & precision results

Opinion Case	Precision
Correct	0.72
Named entity not fully extracted	0.08
Facet not fully extracted	0.18
Generic	0.12
Named entity incorrect	0.13
Facet incorrect	0.12
Relation incorrect	0.09

Table 3.9: Breakdown of judges

3.4.4.3 Analysis of Incorrect Opinions

Table 3.8 shows the precision values for each source as well as the overall accuracy. We first analyzed the reasons for the incorrect opinions. Recall that the judges could indicate that an opinion is incorrect if one or more of the named entity, the extracted facet or the relation was incorrect. We found that of all the opinions that were judged incorrect, only 9% were due to the relation. We conclude then, that our process of constructing the support and oppose lexicon which consists of various patterns for support and oppose, is reasonably robust. We turned our attention to the extraction of named entities and facets – both of which contributed in almost equal measure to incorrect opinions. In the case of named entity extraction, recall that we identify the subject of the sentence as the opinion holder and this is then verified by Yago. However, in cases where the subject cannot be identified by the parser, we make use of a heuristic that the closest noun phrase to the polarized pattern which is identified by Yago as a person, is deemed the opinion holder. We found that this heuristic is not very robust. For example, we found many sentences like the following: "Prime Minister David Cameron met Mr Gates *in Downing Street on Monday and restated* his support *for* American strategy in Afghanistan". *While the facet "American strategy in Afghanistan" is correctly identified, the named entity closest to the polarized pattern is "Mr. Gates", who in this case, is not the opinion holder.*

3.4.4.4 Analysis of Correct Opinions

Table 3.8 shows the overall precision to be about 72%. However, the percentage of opinions which were deemed fully correct was about 41%. In the remaining cases, around 12% of opinions were identified as having very generic facets. For example, from the

Pattern	#Evaluated	Accuracy%	Pattern	#Evaluated	Accuracy%
agrees to	158	69.0	her opposition to	15	86.7
acknowledged that	84	39.3	stood against	15	86.7
he supports	73	80.8	her support of	14	78.6
he accepted	71	67.6	have supported	13	84.6
voted against	70	80.0	refuses to	12	58.3
has supported	62	88.7	voted on	12	8.3
campaigned for	61	90.2	voting on	11	18.2
out against	59	76.3	his support of	11	9.1
he supported	58	72.4	his approval for	10	80.0
opposed to	54	61.1	wants to change	8	100.0
voted for	53	81.1	his disapproval of	8	87.5
voted to	51	56.9	not support	9	100.0
has criticized	48	91.7	urge	7	62.5
his support for	45	82.2	has sharply criticized	7	85.7
he opposed	45	71.1	proposed changes to	7	85.7
has opposed	44	86.4	shift away from	6	100.0
supports	42	78.6	calls for an end to	6	83.3
had supported	39	76.9	have opposed	6	83.3
opposes	38	89.5	lifted a ban on	7	85.7
to overhaul	38	50.0	lobbies for	7	57.1
his opposition to	36	83.3	is a believer in	5	100.0
authorizes	35	68.6	continues to attack	5	60.0
had opposed	32	81.3	he criticizes	5	60.0
called for an end to	32	78.1	his approval of	5	40.0
stand against	31	71.0	looking to change	5	40.0
he agreed to	30	33.3	wants to amend	5	40.0
in favor of	28	78.6	he agreed on	4	100.0
her support for	27	81.5	his opposition of	4	100.0
vote against	26	76.9	have sponsored	4	75.0
campaigned against	25	80.0	his approval to	4	75.0
voting for	25	52.0	would overhaul	4	75.0
voting against	24	75.0	proposes changes to	4	25.0
his criticism of	22	68.2	her criticism of	3	100.0
his support to	21	71.4	lifted restrictions on	3	100.0
was against	21	66.7	his criticism to	3	66.7
overhaul of	20	65.0	comes out in support of	2	100.0
votes against	20	65.0	has strongly criticized	2	100.0
is a supporter of	19	84.2	her approval of	2	100.0
he criticized	19	78.9	her disapproval of	2	100.0
voting to	19	47.4	his approval on	2	100.0
agrees on	18	77.8	supports a repeal of	2	100.0
has sponsored	15	93.3	decided to end	1	100.0

Table 3.10: Accuracy per polarized pattern

*sentence "*George Clooney *on why* he supports the protests*", the facet extracted was "the protests" which is clearly too generic since it does not identify which protest. A further 18% of the opinions were identified as not having their facet correctly ex-*

tracted. The main difficulty here is that many sentences are very long and difficult to parse. Since we rely solely on the dependency parser to extract these facets, our extraction is only as good as the parser. As an example, consider the sentence, "Ivan Rojas said in Colombia that he supports Chavez's plan to secure the release of his sister." The correct facet is "Chavez's plan to secure the release of his sister", while the actual extraction is only partial. Finally, around 8% of opinions were deemed correct, but the named entity was not fully or correctly extracted. On further investigation, we found that the main reason for this was our heuristics as explained in the previous paragraph.

3.4.4.5 Accuracy of Polarized Patterns

The accuracy of support opinions is slightly less than that of oppose opinions due to the polarized patterns that were used and the parsing of sentences (refer to Table 3.8. To understand this, refer to Table 3.10 which shows both the number of opinions extracted by a polarized pattern as well as the accuracy of the opinions. Note that a large number of support opinions were extracted based on the pattern "acknowledged that", but the accuracy of these opinions were small. This was the main reason for the reduced overall accuracy of the support opinions. To investigate why this was the case, we looked further at the breakup of the evaluations. We found that for the opinions derived from "acknowledged that", that were judged incorrect, 100% were because of incorrect facet extraction. When we further investigated the actual sentences, we found them to be generally long and difficult to parse. For example, the following sentence was extracted: "Kerry acknowledged that the bill would raise energy prices, ...". While the named entity "John Kerry" was correctly identified, the facet was incorrectly identified as "energy". We noticed that this has to do with the nature of the polarized pattern itself. The pattern "acknowledged that" is usually accompanied by another verb phrase (this also true for "agreed to", but not "supports"). This leads the parser to incorrectly identify the object of the verb phrase to be the facet, instead of the verb phrase itself. We found several other sentences of a similar nature and conclude that the parsing should be more robust since our facet extraction depends on it.

3.4.4.6 Analysis of Online Sources

We additionally notice a variation in the accuracy of opinions from different sources. Most noticeable are opinions extracted from BBC and Wikipedia, both of which are at the bottom. One of the reasons we specifically chose news sources for opinion extraction *is because of the writing style. The writing style in news articles typically tends to be opinionual, consists of short sentences and correct grammar. However, in the case of the BBC, we noticed that most of the result snippets returned were from user blogs under the BBC site and online comments and not necessarily from the articles themselves. This difference in style resulted in lower extraction accuracy. For Wikipedia, one of the main problems was in the extraction of the correct named entity. In general, Wikipedia articles are very long and detailed, not necessarily focused on a single theme (this is in contrast to news articles which are short and focused). Moreover, Wikipedia articles typically contain references to many different people (especially in biographical pages). Our simple heuristic of finding the nearest named entity reference (in the absence of named entity in the sentence itself) fails most often for Wikipedia. And this resulted in the lower accuracy.*

3.4.5 Results for Focused Sample

We created a focused sample by manually identifying hot topics in each of the geo-political regions Africa, Asia, Europe, Middle East, and USA, and combining them with prominent politicians from these areas. Table 3.11 shows the evaluation results. For each cell of the table, we list the numbers of support (+) and oppose (−) opinions found in our corpus, and give the precision ($p = 0.93$) *based on manual assessment by one of our judges. A subset of the evaluated opinions are shown in Table 3.12.*

	US: offshore drilling	Africa: election in Cote d'Ivoire	Mid-East: NATO action in Libya	Europe: EU aids for Greece	Asia: nuclear power plants
Barack Obama	10+, 7−, $p = 1.0$	2+, $p = 1.0$	4+, $p = 1.0$		2+, 2−, $p = 1.0$
Newt Gingrich	1+, $p = 1.0$		1−, $p = 1.0$		1+, $p = 1.0$
Kofi Annan		2+, $p = 1.0$	1+, $p = 1.0$		
Laurent Gbagbo		2−, $p = 1.0$			
Amro Mousa			1+, $p = 1.0$		
Binyamin Netanyahu					1−, $p = 0.0$
Angela Merkel				2+, 1−, $p = 1.0$	2+, 2−, $p = 0.75$
Nicolas Sarkozy		2+, $p = 1.0$		1+, $p = 1.0$	1+, $p = 1.0$
Naoto Kan					1+, $p = 1.0$
Manmohan Singh					1+, $p = 1.0$

Table 3.11: Results of the focused sample

3.4.6 Results for Ground-Truth Set

As the previous sample-based evaluation does not allow us to make any recall estimates, we also compared a subset of our automatically compiled opinions to a ground-truth dataset with largely aggregated and thus quasi-objective opinion polarities. To this end, we obtained the data on eight US politicians (from both democratic and republican parties, e.g., Obama, McCain, etc.) on 19 controversial topics, from the two web sites `procon.org` *and* `ontheissues.org`*. The topics were major themes such as abortion, same-sex marriage, death penalty, illegal immigration, etc.. We aligned opinions triples in our database with ground-truth triples by first matching the named entities and then matching the ground-truth topics against i) the facets alone, ii) the Debatepedia topics of the facets. In the later case, we select the two topics with the highest Jaccard similarity to the ground-truth topic, based on the terms in the topics' facet strings.*

Table 3.13 shows the results for precision and recall of this evaluation study. Note that the recall for facets is lower than for the Debatepedia topics. This is caused by the

1. I remember thinking just how odd and out of character it was that **Barack Obama** had announced his approval for *more offshore drilling*
2. **Obama** has lifted a ban on *deepwater oil drilling in the Gulf of Mexico* and set new safety conditions
3. The **Obama** administration defended *six-month moratorium on U.S. deepwater offshore drilling* in court on Wednesday
4. **Obama** has opposed new *offshore drilling*
5. **Mr. Obama has** is softening his opposition to *offshore oil drilling*
6. More likely, **Obama** believed - and continues to believe, since even now he has not withdrawn his support for *offshore drilling*
7. **US President Barack Obama** has lifted a ban on *deepwater drilling in the Gulf of Mexico* that was imposed after the BP oil spill
8. **Mr. McCain** also supports *ocean drilling*
9. A day after Senator **Barack Obama** said he could support *broad energy legislation even though it would permit offshore oil drilling*
10. **Obama** was against *ending the current ban on offshore drilling*
11. **Obama** opposes new *offshore drilling* at home and has derided nuclear power
12. **Mr. Obama** said several times during his presidential campaign that he supported *expanded offshore drilling*
13. **Barack Obama** had announced his approval for *more offshore drilling*
14. **Barack Obama** told a Florida newspaper today he is not against all *offshore drilling for new oil resources*
15. **President Barack Obama** said he continues to support *offshore oil exploration* "if it can be done safely,"
16. **Obama** said Friday that he would be willing to compromise on his position against *offshore oil drilling* if it were part of a more overarching strategy to lower energy costs
17. **Barack Obama** stood his ground Wednesday in opposing what he calls the "scheme" of *offshore drilling*
18. **Obama's** support for *nuclear power*
19. **Barack Obama**, he opposes *promoting nuclear power*
20. **Obama** opposes *new offshore drilling* at home and has derided nuclear power
21. **Barack Obama** on Tuesday criticized *his rival John McCain's proposal to encourage the building of 45 new nuclear reactors by 2030*
22. **Obama** offers support to *Ivory Coast leader Ouattara*
23. **President Obama** have strongly supported the results of *Cote d'Ivoire's democratic election*.
24. **Obama** strongly defends *US military action in Libya*
25. **President Obama** supports *military action against Moammar Gadhafi* if he continues to attack Libyan rebels
26. I am disappointed that **Barack Obama** who said that he opposed use of force in Iraq will not oppose *this horrible military action against Libya*
27. **Obama** authorizes *military action against Libya*
28. **Gingrich** mounts campaign to support *domestic oil drilling*
29. **Gingrich** has been a staunch supporter of *nuclear power*
30. **Gingrich** said that he didn't favor *military intervention in Libya*
31. **Laurent Gbagbo** refuses *to cede power in Ivory Coast*
32. **Sarkozy** in Ivory Coast to support *Ouattara*
33. **Sarkozy** gave the go-ahead for his troops to join a UN operation against *forces loyal to Laurent Gbagbo*
34. **Sarkozy** lobbies for *nuclear power*
35. **Sarkozy** urges *EU to help Greece*
36. **Kofi Annan** supports *the many people in Cote d'Ivoire* who are standing up against the current injustice and repression
37. Former UN Secretary **General Kofi Annan**, originally supportive of *the mission in Libya*
38. **Sarkozy** approves *EU-IMF Aid Deal for Greece*
39. **Merkel** for her support of *a $145 billion joint E.U. and IMF bailout for Greece*
40. **Merkel** refuses to *commit to larger aid plan for Greece*

Table 3.12: Examples from the focused sample

	Facets	Debatepedia
Precision	0.98	0.81
Recall	0.30	0.42

Table 3.13: Results for experiment with objective ground truth

inherent difficulty of matching fine-grained facets against coarse-grained main topics. For this situation, a recall of 30 percent is a satisfactory result. We further investigated the reason behind the very high precision of both the focused set and the ground-truth set compared to the precision of the random set. Attributed to the nature of the topics in these sets, 96% of the matched triples were retrieved using the patterns: "support", "against", "voted for", "voted against". Sentences including these patterns are parsed with high accuracy. On the otherhand, different patterns with different parsing accuracies caused the low overall precision of the random set.

3.5 Related Work

For gathering statements that connect individual entities with opinion expressions, we make use of techniques for information extraction from text and Web sources. [37, 108] give overviews of state-of-the-art methods. Our approach builds specifically on the pattern-fact duality principle that goes back to [25, 3] and allows us to bootstrap the acquisition process with very few seeds. We customize this general framework to collect opinionated cues rather than general relational patterns. Although identification of features is in some sense a standard entity recognition problem, an opinion extraction system would be mostly interested in features for which associated opinions exist; similarly, an opinion holder is not just any named entity in a news article, but one that expresses opinions.

Opinion mining (aka. sentiment mining) is a richly researched topic that gained a lot of attention in recent years for business intelligence, smart advertisements, marketing campaigns, etc.. The standard model for these mining tasks is to identify objects (e.g., cameras or movies), facets (aspects, features) of these objects on which a sentiment is expressed (e.g., the camera's ease of use, or the movie's special effects), and a polarity of the opinionated statement (positive or negative), and sometimes also an opinion strength for quantifying the polarity. Typically, these cues are aggregated to form opinion profiles, *based on many opinions like customer reviews or postings in discussion forums. [94, 81] are excellent surveys on the methodologies for doing this kind of opinion analysis. Our work uses specific techniques from this state of the art, but adapts them to our setting and combines them with new methods. Moreover, the opinion holder is crucial to our task, discriminating our goal from that of traditional sentiment mining: we are interested in the individual entities that express opinions rather than merely aggregating over many individuals.*

3.5.1 Features Extraction

Feature-based sentiment analysis research is mainly based on online product reviews. Some approaches rely on the linguistic heuristic that features are usually expressed as noun phrases. For example in [57] they identify frequent features through association mining. Some heuristics are used to prune noun phrases in order to remove i) multi-word candidates in which the words do not appear together in a certain order, and ii) single-word candidates for which subsuming super-strings have been collected (for example, "life" is discarded in favor of "battery life"). In [132] they use part of speech patterns to select from the extracted set of noun phrases, those which represent features. Other approaches are based on the idea that an opinion should have a target either an entity or a feature of an entity. A double propagation approach is proposed in [103]. They use this dependency between the opinion and the target to extract both features and opinion words. Their approach is a bootstrapping method that receives as input a set of seed opinion words, and based on a predefined set of dependency grammer rules they extract the associated opinions or targets. In [115], a model which is based on bipartite graph is proposed for features extraction. It exploits the mutual reinforcement relationship between the features and the opinions words. Specifically, they use the co-occurrence of a feature and an opinion word pair in a sentence. Their algorithm iteratively clusters the set of features and the set of opinion words separately, but before clustering each set, the clustering results of the other set is used to update the pairwise weight of the set.

3.5.2 Polarity Identification

For the identification of the sentiment or the opinion polarity of the extracted features, many works utilize machine learning based classifiers for the sentiment classification of texts. They deal with this problem as a classification problem since it is concerned *with two opposing subjective classes (see [95, 94, 122, 135] for examples of classifying product reviews, and [119, 15, 26, 86] for examples of classifying political text). A target-dependent polarity classification approach is described in [59]. In particular, they address target-dependent sentiment classification of tweets. They incorporate syntactic features to distinguish texts used for expressing sentiments towards different targets in a tweet. In addition, they take the related tweets of the current tweet into consideration by utilizing graph-based optimization. Other works use opinion lexicons of words and phrases in order to identify polarities. Such lexicons are constructed either manually or automatically [36, 103, 40, 18].*

3.5.3 Joint Feature-Polarity Analysis

Other works propose joint feature-polarity statistical models [87, 120, 121, 85, 124, 109, 75, 45] based on topic models (LDA) [22]. For example in [109], they provide a mechanism for review content aggregation that identifies fine-grained product properties across reviews (e.g., battery life for electronics or pizza for restaurants) as well as capturing attributes of these properties, namely aggregate user sentiment. For this task, they propose an approach that jointly analyzes the whole collection of product review snippets, induces a set of learned properties, and models the aggregate user sentiment towards these properties. They capture this idea using a Bayesian topic model where a set of properties and corresponding attribute tendencies are represented as hidden variables. The model takes product review snippets as input and explains how the observed text arises from the latent variables, thereby connecting text fragments with corresponding properties and attributes. Their model is a variation of LDA but with seeds for sentiment words as priors. It also has a Hidden Markov Model for modeling the sequence of words with types (feature word, sentiment word, and background word). They approximate the full model posterior using variational inference [77]. Opinions at the collection level with each collection on a topic coming from a different perspective are examined by the work described in [45]. A latent topic model is devised to discover the common topics across all the perspectives. For each topic, the opinions from each perspective are summarized. In their model, the opinion generation process is separated from the topic term generation process. They assume that topics are expressed through noun words, and opinions are conveyed through adjective, verb and adverb words. In this model, it is assumed that topics are shared among all the documents, regardless the perspective of the document. Therefore, the topic words are drawn from the shared topic word distribu-

tion. On the other hand, the opinions from different perspective could be different. Thus, the opinion words are drawn from the topic-opinion distribution conditioned on the perspective. Specifically, the topic word (t) is modeled by a shared LDA across perspectives. The opinion word (o) is drawn conditioned on the topic (x) which is uniformly sampled from the topics learned from the topic words in document d.

3.5.4 Opinion Holders Extraction

There are different systems that extract opinion holders from news articles and political debates [15, 16, 118]. An approach is described in [29] that combines Conditional Random Fields (CRFs) [74] and extraction patterns. A CRF model is trained on a collection of lexical, syntactic, and semantic features. Extraction patterns are learned to provide semantic tagging as part of the semantic features. Some other works take into consideration the opinion expression for the identification of opinion holders. For example, an integer linear programming approach is employed to handle the joint extraction of entities and relations using global inference based on constraints [28]. Also in [20], an approach which is based on semantic parsing is described. It marks the semantic constituents of sentences (e.g., "agent" or "proposition"). They utilize opinion words which are automatically learned by a bootstrapping approach, in order to refine the semantic roles to identify propositional opinions. Other systems start by identifying opinion expressions, and then proceed to the analysis of the opinions, including the identification of opinion holders. For example, structural features from a syntactic parse tree are selected to model the long-distance, structural relation between a holder and an opinion expression [69]. Our approach uses similar techniques as described in [69], but for matching names in text sources against canonical entities, we leverage existing knowledge bases like DBpedia, Freebase, or Yago. We specifically make use of the means relation that Yago [116] provides for individual entities and their lexical name variations. This information is in turn derived from anchor texts and redirects of Wikipedia.

3.6 Summary

In this chapter, we described our approach to acquire direct opinions and to extract triples-contexts (opinion targets, opinion holders and their opinions) from Web result snippets using an initial set of seed patterns. Our evaluation showed an overall precision of 72%.

Chapter 4

Indirect Opinions

4.1 Introduction

Political analysts are often interested in individual and brief statements, as quoted in news media, and their pro/con polarity *with regard to* fine-grained debates *such as "deporting illegal immigrants" or "immigration amnesty".* Quotations *are a prominent form of highlighting opinions in newspapers, broadcast stations, online forums, and all kinds of social media [16]. Different news media select different quotations of the same person to express or amplify their specific slants on controversial topics. If one could properly analyze this wealth of opinionated statements, this would enable a much more fine-grained map of the media landscape and its political orientations. The opinions expressed in quotations that appear in news media usually reflect their opinion holders beliefs or arguments with respect to the topics of the quotations.*

In our work we focus on quotations as a source of indirect opinions which were defined in Chapter 2. The problem addressed in this chapter is to automatically classify quotations onto fine-grained topics of controversial nature, and to identify a pro/con polarity for each fine-grained topic.

In the rest of this chapter, we first describe our computational model in Section 4.2. In Section 4.3, we describe our approach for pro and con opinion classification. Section 4.4 presents our experimental evaluation against a variety of baselines. Section 4.5 positions our contributions with regard to related work. We conclude with Section 4.6.

4.2 Computational Model

Our model considers quotations (short documents), debates (fine-grained topics), and polarities (pro/con stances).

4.2.1 Quotations

A quotation in news media is a relatively short text. The quoted person or party, can refer to several topics. Table 4.1 shows examples with quotations on "immigration" and on "the conflict in Syria". We are interested in identifying the pro/con stances not just on the broad issue of "immigration" or "the conflict in Syria" but on fine-grained topics: for example, ***pro*** *"Mexican border fence",* ***con*** *"deporting illegal immigrants", for the fourth quotation, and* ***con*** *"Arming Rebels in Syria",* ***pro*** *"Departure of Asaad", for the sixth quotation.*

Definition 4.1 ***Quotation***
A **quotation** *$\mathcal{Q}$ is a short* **document** *(for example, a paragraph or a sentence) that i) addresses a set of topics $\mathcal{D}_{all} = \{\mathcal{D}_1, ..., \mathcal{D}_k\}$, and, ii) expresses an opinion (pro or con) on each $\mathcal{D}_i$.*

Challenging issues with quotations. *We cannot simply map a quotation into pro or con based on the identity of the speaker and her/his overall attitude derived from an entire corpus (e.g., Michele Bachmann is against immigration). At the fine granularity that we consider in this work, the same speaker may express different pro/con attitudes on specific topics at different times and in different contexts.*

The problem is much more demanding than traditional forms of sentiment mining for various reasons:

1. Fine granularity: *the number of debated topics that the classifier must consider is potentially high, in the thousands rather than the usual tens;*

2. Brevity of statements: *quotations are usually short texts (often only a single sentence), so whatever linguistic features are used tend to be sparse;*

3. Topic-dependent polarity: *the same quotation can have different polarities for different fine-grained topics.*

Quotation (1)*:** "The alternatives to the Specter bill are senseless. The enforcement-only approach – building a 700-mile wall and engaging in a campaign of mass deportation and harassment to rip 12 million people from the national fabric would destroy families and weaken the economy." ***con Mexican border fence ***con*** deporting illegal immigrants
Quotation (2)*:** "I know fencing helps secure our nation's borders because criminal activity in every statistical category has been eliminated or decreased since we built the border fence in San Diego County. What was once a porous border, susceptible to illegal aliens, drug trafficking and terrorism, is now the standard mode in preventing drug smugglers from bringing narcotics into our neighborhoods and allowing border enforcement personnel to reinforce areas of greater need." (**Duncan Hunter**) ***pro Mexican border fence
Quotation (3)*:** "I think that the wall could help with the economy" ***pro Mexican border fence
Quotation (4)*:** **Rick Santorum** has said that the fence should be finished, but that mass deportations are not the answer. "Until we build that border, we should neither have storm troopers come in and throw people out of the country nor should we provide amnesty." ***pro Mexican border fence ***con*** deporting illegal immigrants
Quotation (5)*:** "The Iranians and the Russians are providing Bashar Assad with weapons. People that are being massacred deserve to have the ability to defend themselves." **McCain** said. ***pro Arming Rebels in Syria ***con*** Arming Assad Regime
Quotation (6)*:** "We made a decision not to provide lethal assistance at this point. I know others have made their own decisions. But I think it's very important right now that everybody focus on a smooth and responsible political transition," **Panetta** said. ***con Arming Rebels in Syria ***pro*** Departure of Asaad

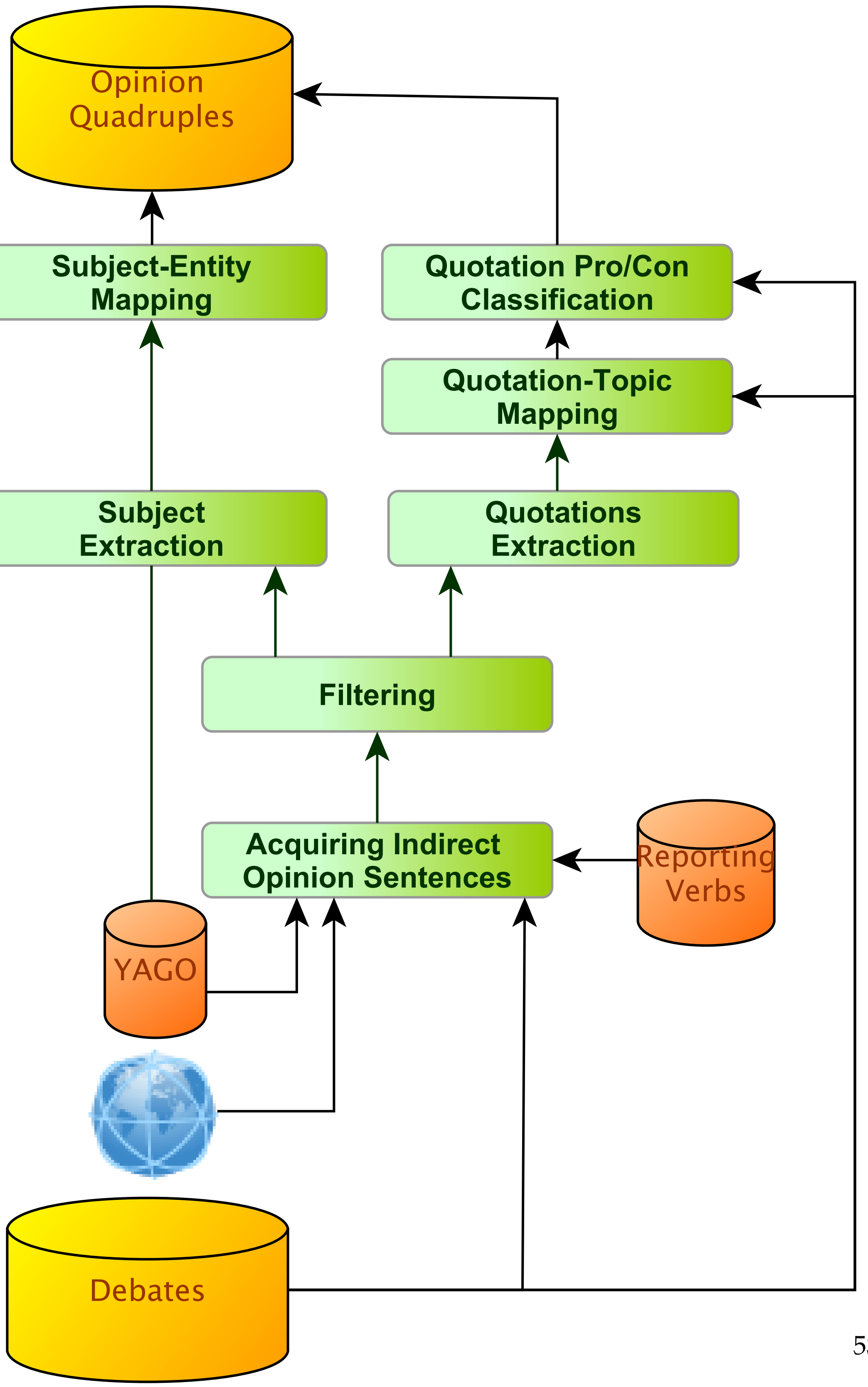
Table 4.1: Quotations about immigration and about the conflict in Syria

Acquiring quotations. *In order to collect quotations related to controversial topics (e.g. "gun control", "gay rights", and "capital punishment"), we first query the Web– limiting the sites to online news media– for articles about topics selected from a manually constructed list of controversial topics. The returned articles are then queried for specific reporting patterns such as "said", "says", "mentioned", etc., in order to collect quotations. Note that, while we broadly know the topic of the quotation, our aim is to extract* fine-grained topics*. Moreover, quotations may contain multiple ⟨opinion, fine-grained topic⟩ pairs, each of which need to be extracted. In addition we need to identify the opinion holder. We follow the approach described in Section 3.3.1 of Chapter 3 in order to extract the quoted entities which represent the opinion holders. The* opinion quadruple *set of the extracted opinions is constructed from the identified quoted entities (the opinion holders), the identified pro/con polarities, the identified fine-grained topics, and the contexts represented by the quotations. See Figure 4.1 for an overview of the process we follow to extract indirect opinions (or quotations) from news media.*

4.2.2 Debates

Our approach which we will describe in more detail in Section 4.3, maps quotations onto one or more topics in a category system of political debates. We use the fine-grained categories of `debatepedia.org` *as our target; there are about 1,700 such categories, called* debates *in debatepedia. Debatepedia is the Wikipedia of debates - an encyclopedia of pro and con arguments and quotations on critical issues. It utilizes the same wiki technology powering Wikipedia to centralize arguments and quotes found in editorials, op-eds, political statements, and books into comprehensive pro/con articles. For example, the category "Immigration" in Debatepedia contains the debates "700 mile US Mexico border fence", "Ban on renting to illegal immigrants", "DREAM Act", "Deporting illegal immigrants", "Drivers licenses for Illegal immigrants", "Citizenship for illegal immigrants", "Voting rights for legal immigrants", etc.. Each of these debates has pro an con arguments and quotations.*

Definition 4.2 *Debate*
A **debate** *denoted as $\mathcal{D}$ is a fine-grained* **topic** *of a controversial nature. A debate comprises of a number of documents, each of which presents opinions of exactly one of two opposing polarities among the opinion holders:* pro *or* con*. We use the words debate*


Opinion Quadruples
Subject-Entity Mapping
Quotation Pro/Con Classification
Quotation-Topic Mapping
Subject Extraction
Quotations Extraction
Filtering
Acquiring Indirect Opinion Sentences
Reporting Verbs
YAGO
Debates

and topic interchangeably.

4.3 Language-Model-Based Opinion Classification

The topics and pro/con opinions in a quotation are given in latent, textual form and are a priori unknown. The classification model that we pursue in this work maps a quotation onto one or more topics and a pro/con polarity for each selected topic. The topics are explicitly given by a directory such as Debatepedia. For example, for the input texts in Table 4.1, the output of our method is the topic/polarity pairs shown at the bottom of each quotation. The classifier's targets, the debates (categories) in Debatepedia, come with articles and user discussions. We define statistical language models (LMs), with judiciously chosen features (including bigrams) for each debate, and then use a scoring function based on the query likelihood as a similarity measure that is fed into different kinds of unsupervised or supervised classifiers (kNN, SVM, LDA). The problem of high sparsity in the debates themselves is addressed by smoothing the debates' LMs via thematically related debates.

Our approach constructs statistical language models for each debate. Then, we compute a score based on the query likelihood [136, 33] treating a debate as a document and a quotation in the role of a query.

However, estimating an LM with just unigrams or bigrams is not sufficient for two reasons: (1) the brevity of quotations, (2) our need for classifying the quotation onto a topic and a polarity.

To overcome the difficulty caused by the brevity of quotations and sparseness of features, we have devised a method of quotation expansion *that harnesses thesauri like WordNet [46]. We use synonyms and antonyms (i.e., words for opposite senses, e.g., "censorship" or "regulation" as antonyms of words like "neutrality" or "freedom") to conceptually expand the text of a quotation. This technique is based on the intuition that opinions that agree with one of the two stances tend to use synonymous phrases, whereas opposite opinions tend to use antonymous wording. This approach leads to a novel form of enriched feature space: a quotation is then represented by an expanded entailment/contradiction language model.*

For the second issue, we devise a feature model that distinguishes between topical terms and sentiment terms. The expansion technique treats these two groups of terms differently. The LMs are enriched by considering pairs of topical and sentiment terms as *features. This coupling of the two aspects of a quotation improves the classification onto topics and polarities.*

4.3.1 Features Model

4.3.1.1 Topic and Sentiment Terms

Definition 4.3 ***Topic Term***
A **topic term** *is a term which describes a topic. We assume nouns to be topic terms.*

For example, quotation (1) in Table 4.1 has the topic terms "wall", "campaign", "economy", "deportation", etc., as shown in Table 4.2.

Definition 4.4 ***Sentiment Term***
A **sentiment term** *is a term which describes an opinion. We assume verbs, adjectives and adverbs to be sentiment term.*

For example, quotation (1) in Table 4.1 has the sentiment terms "destroy", "weaken", "senseless", etc., as shown in Table 4.2.

Term	Part of Speech	Type	Synonyms	Antonyms
fence	noun	topic term	wall, border	-
help	verb	sentiment term	aid, assist, support, encourage, not hurt, strength	not help, destroy, harm, hinder, stop, weaken
deportation	noun	topic term	banishment, displacement, exile, expatriation	approval, permission

Table 4.2: Synonyms and antonyms of terms

4.3.1.2 Unary and Binary Features

Definition 4.5 ***Unary Feature***
A **unary feature** *is denoted as* $\langle \mathrm{u} \rangle$ *where* u *is either a topic term or a sentiment term.*

For example, quotation (1) in Table 4.1 has the following unary features: $\langle$*wall*$\rangle$, $\langle$*campaign*$\rangle$, $\langle$*economy*$\rangle$, $\langle$*deportation*$\rangle$, $\langle$*destroy*$\rangle$, $\langle$*weaken*$\rangle$, *etc..*

Definition 4.6 ***Binary Feature***
For a given quotation $\mathcal{Q}$, *let* $\mathcal{Q}^{\mathsf{T}}$ *and* $\mathcal{Q}^{\mathsf{S}}$ *denote the set of its topic terms and sentiment terms respectively. A* **binary feature**, *denoted* $\langle \mathsf{t}, \mathsf{s} \rangle$, *consists of* $\mathsf{t} \in \mathcal{Q}^{\mathsf{T}}$ *and* $\mathsf{s} \in \mathcal{Q}^{\mathsf{S}}$, *such that,* t *and* s *are connected by a dependency relation. The dependency relation is determined by parsing the sentence in* $\mathcal{Q}$ *in which they co-occur using a dependency parser*[1].

For example, quotation (1) in Table 4.1 has the following as binary features: $\langle$*wall, weaken*$\rangle$, $\langle$*deportation, destroy*$\rangle$, $\langle$*weaken, economy*$\rangle$, *etc..*

4.3.2 Quotations Expansion

Our approach of pro and con classification is built on the intuition that opinions which are in agreement with each other have expressions which are in agreement to each other, while opinions which disagree have expressions which are in disagreement.

For example in Table 4.1 quotation (1) which has the expression "…and weaken the economy." is in disagreement *with the expression in quotation (3) "…could help with the economy", while the expression "…mass deportation is not the answer" in quotation (4) is in* agreement *with the expression "engaging in a campaign of mass deportation…would destroy families" in quotation (1).*

In order to capture the notion of agreement *and* disagreement *for a given quotation, we focus specifically on the binary features of the quotation. That is, the topic and sentiment term pair* $\langle \mathsf{t}, \mathsf{s} \rangle$ *which are in a dependency relationship with each other. The key idea that we propose is to expand the topic term and the sentiment term with both their synonyms as well as their antonyms (see Table 4.2). For this expansion, we use the WordNet thesaurus [46], which gives synonyms and antonyms for many concepts. In order to map a word observed in a quotation onto its proper word sense, that is, the WordNet concept denoted by the potentially ambiguous word, we use the most-common-sense heuristics which has been used effectively in many applications [100].*

Definition 4.7 ***Term Synonyms and Antonyms***
For a topic term t_i, *the set of* ***topic term synonyms*** *is denoted as* $\mathcal{T}_i$, *and the set of*

[1] `nlp.stanford.edu`

topic term antonyms *is denoted as* $\bar{\mathcal{T}}_i$*. Analogously,* $\mathcal{S}_i$ *and* $\bar{\mathcal{S}}_i$ *denote the sentiment term synonyms and antonyms of a* ***sentiment term*** s_i*, respectively.*

4.3.2.1 Agreement and Disagreement Features

Let t^+, t^- *denote a synonym and antonym of a topic term respectively. Analogously,* s^+ *and* s^- *denote a synonym and antonym of a sentiment term. The possible expansions of a binary feature* $\langle t, s\rangle$ *are the pairs* $\langle t^+, s^+\rangle$, $\langle t^-, s^-\rangle$, $\langle t^-, s^+\rangle$, $\langle t^+, s^-\rangle$*. The first two are in agreement with the original feature* $\langle t, s\rangle$*, while the last two are in disagreement.*

Quotation	"The *wall* could **help** with the *economy*"
Topic terms	wall, economy
Topic term synonyms	{wall, fence, border}, {economy, saving}
Topic term antonyms	{}, {spending, expend}
Sentiment terms	help
Sentiment term synonyms	{help, assist, support}
Sentiment term antonyms	{destroy, weaken, not help}
Original feature	$\langle \mathrm{economy}, \mathrm{help}\rangle$
Features in agreement	$\langle \mathrm{economy}, \mathrm{assist}\rangle$, $\langle \mathrm{economy}, \mathrm{support}\rangle$
Features in disagreement	$\langle \mathrm{economy}, \mathrm{destroy}\rangle$, $\langle \mathrm{spending}, \mathrm{support}\rangle$

Table 4.3: Features in agreement/disagreement with original features

As an example, consider the quotation in Table 4.3 and the binary feature ⟨economy, help⟩. A synonym for the topic term "economy" is "saving" while its antonyms could include "spending" and "expend". Similarly, for the sentiment term "help", synonyms include "support" and "assist". while it's antonyms are "destroy", "weaken". Therefore, the expanded binary features include ⟨economy, weaken⟩ ($\langle t^+, s^-\rangle$*), which is in* disagreement *with the original feature, as well as ⟨saving, assist⟩ (*$\langle t^+, s^+\rangle$*), which is in* agreement *with the original feature. Figure 4.2 shows more examples of these binary features.*

Definition 4.8 *Agreement Features*
For a given binary feature $\langle t_i, s_i\rangle$ *present in the quotation, we define the set of* **agreement features** *as* $\mathrm{AF} = \{\langle t_i', s_i'\rangle | t_i' \in \mathcal{T}_i, s_i' \in \mathcal{S}_i\} \cup \{\langle t_i', s_i'\rangle | t_i' \in \bar{\mathcal{T}}_i, s_i' \in \bar{\mathcal{S}}_i\}$

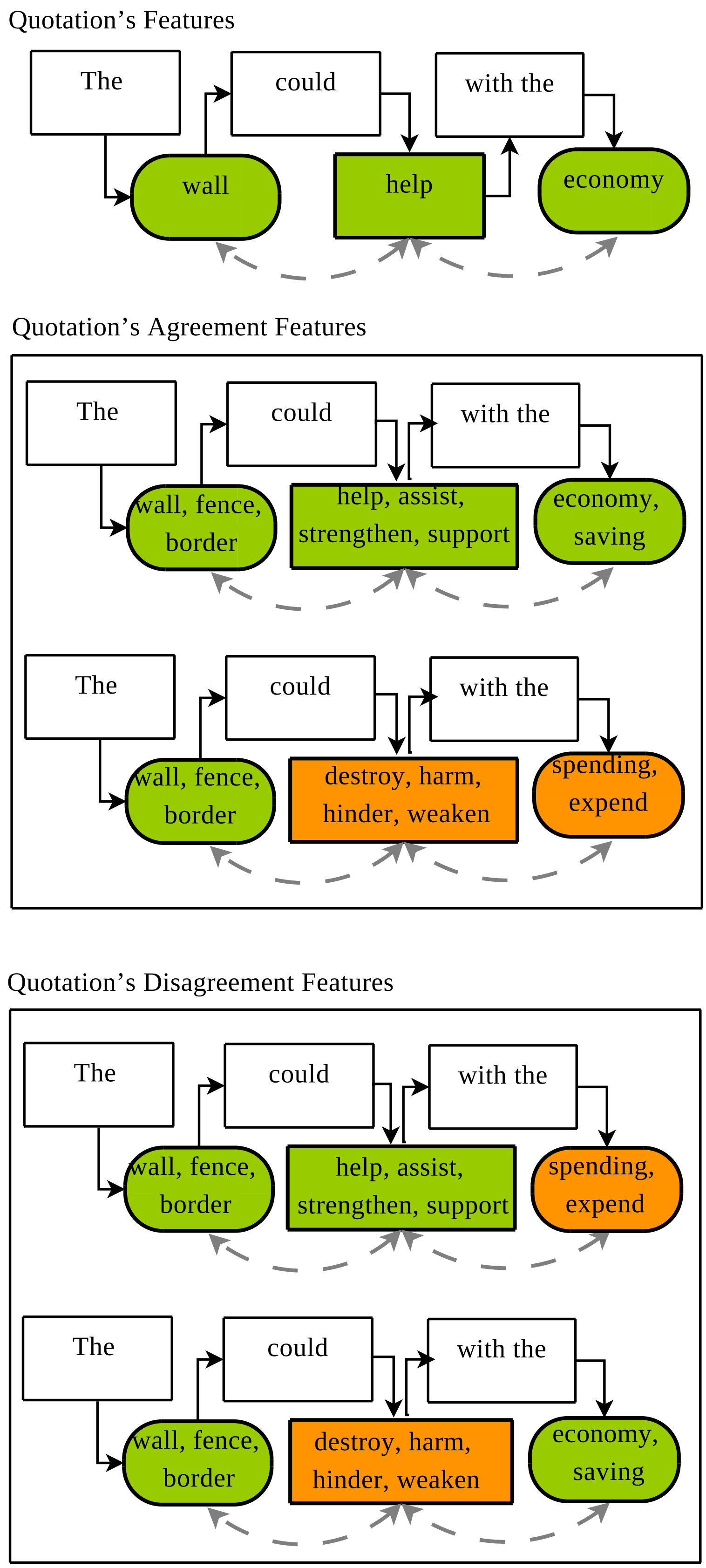


Figure 4.2: Expanding quotation by agreement/disagreement features: ellipse (topic term), box (sentiment term), green (synonyms), red (antonyms)

Definition 4.9 ***Disagreement Features***
For a given binary feature $\langle t_i, s_i \rangle$ *present in the quotation, we define the set of* **disagreement features** *as* $DF = \{\langle t_i', s_i' \rangle | t_i' \in \mathcal{T}_i, s_i' \in \bar{\mathcal{S}}_i\} \cup \{\langle t_i', s_i' \rangle | t_i' \in \bar{\mathcal{T}}_i, s_i' \in \mathcal{S}_i\}$

4.3.3 Language-Model-Based Ranking for Information Retrieval

The goal of an information retrieval (IR) system [136] is to rank documents given a query based on relevance. The retrieval accuracy of an IR system is determined by the quality of the scoring function adopted. A retrieval function is based on a retrieval model. Over the decades, there have been many empirical studies of models such as the vector-space model with heuristic Tf–idf weighting and document length normalization , and the BM25 (Okapi) retrieval function which is motivated and derived from the 2-Poisson probabilistic retrieval model with heuristic approximations. An interesting class of probabilistic models called language modeling approaches to retrieval have led to effective retrieval functions without much heuristic design. In particular, the query likelihood retrieval function with Dirichlet prior smoothing has comparable performance to the most effective Tf–idf weighting retrieval functions including BM25. Language models have now been applied to multiple retrieval tasks such as cross-lingual retrieval, distributed IR, expert finding, passage retrieval, web search, genomics retrieval, topic tracking, and subtopic retrieval. Before it was applied to retrieval, it had already been used successfully in related areas such as speech recognition and machine translation. In these applications, language models are used to assess what kind of word sequences are more typical according to language usages, and inject the right bias accordingly into a speech recognition system or machine translation system to prefer an output sequence of words with high probability according to the language model.

The term language model refers to a probabilistic model of text (i.e., it defines a probability distribution over sequences of words). The language modeling approach was first introduced in [99]. They proposed a new way to score a document, later often called the query likelihood scoring method. We briefly explain the query likelihood model next.

4.3.3.1 Query Likelihood Model

In the basic language modeling approach proposed by Ponte and Croft, the query is assumed to be a sample of words drawn according to a language model estimated based *on a document (i.e., a document language model). The question now which document language model gives the query the highest probability. Therefore documents are ranked based on the likelihood of generating the query using the corresponding document model. Intuitively, if a document language model gives the query a high probability, the query words must have high probabilities according to the document language model, which further means that the query words occur frequently in the document. Formally, the general idea of the query likelihood retrieval function can be described as follows.*

Definition 4.10 ***Query Likelihood***
Let Q be a query and D a document. The query likelihood of a document is the probability of generating the query given the language model of the document and is denoted as $P(Q|M_D)$.

The score of document D with respect to query Q is defined as the conditional probability $P(Q|M_D)$. That is,

$$\mathtt{score}(Q, D) = P(Q|M_D) \tag{4.1}$$

The language model of a document D can defined as a multiple Bernoulli model [99], or more commonly as a multinomial unigram language model over its set of words V, where the documents are the classes, each treated in the estimation as a separate "language" [52, 88].

Under the multinomial unigram language model, and given a language model M_D of a document D, the likelihood of a query $Q = \{q_1, q_2, \dots q_m\}$, where q_i is a query word, is computed as follows.

$$P(Q|M_D) = K_Q \prod_{w_i \in Q} P(w_i|M_D)^{c(w_i;Q)} \tag{4.2}$$

where, $K_Q = |Q|!/(c(w_1;Q)!c(w_2;Q)! \cdots c(w_{|Q|};Q)!)$ is the multinomial coefficient for the query Q. $c(w_i;Q)$ is the count of word w_i in Q. $|Q| = \sum_{w_i \in Q} c(w_i;Q)$ is the length of Q. $P(w_i|M_D)$ is the probability of word $w_i \in Q$ in the language model M_D where $\sum_{w_i \in V} P(w_i|M_D) = 1$. In practice the multinomial coefficient is left out in the calculations, since, it is a constant for a particular query.

The retrieval problem is now reduced to the problem of estimating the probability $P(w_i|M_D)$ for every word w_i. The maximum-likelihood (ML) estimator is used as follows.

$$P(w_i|M_D) = \frac{c(w_i;D)}{|D|} \tag{4.3}$$

where $c(w_i; D)$ *is the count of word* w_i *in* D.

One problem with the ML estimator is that an unseen word in document D *would get a zero probability, making all queries containing an unseen word have zero probability. Therefore the issue of smoothing the ML estimate is critical. One way to smooth th ML is to interpolate it with a background language model estimated using the entire collection as follows.*

$$P(w_i|M_D) = (1-\lambda)\frac{c(w_i; D)}{|D|} + (\lambda)P(w_i|C) \tag{4.4}$$

where $P(w_i|C)$ *is a collection (background) language model estimated based on* w_i *counts in the entire collection and* $\lambda \in [0, 1]$ *is a smoothing parameter.*

4.3.4 Quotations Classification by Topic

Our classification model of quotations into topics is based on the query likehood model defined over a multinomial model. We first discover the topics covered by a quotation, without regard to their polarities. We estimate a language model for each debate with unary *features of the* topic terms *and then compute the query likelihood as explained below.*

Definition 4.11 ***Debate Unary-Topic Language Model***
The language model of a debate $\mathcal{D}$ *denoted as* $P_{\mathcal{D}}$ *is equal to:*

$$P_{\mathcal{D}}(w) = (1-\lambda)P(w|\mathcal{D}) + \lambda P(w|\mathcal{C}_{\mathcal{D}}) \tag{4.5}$$

where $\mathcal{D}$ *is a debate on a fine-grained topic,* w *is* ***a topic term*** *as defined in Section 4.3.1.1, and* $\mathcal{C}_{\mathcal{D}}$ *is the set of debates in the same category of* $\mathcal{D}$ *in Debatepedia. The parameters of the language model of the debate are estimated using the maximum-likelihood estimator as follows.*

$$P(w|\mathcal{D}) = \frac{c(w; \mathcal{D})}{\Sigma_i c(w_i; \mathcal{D})} \tag{4.6}$$

$$P(w|C_D) = \frac{c(w; C_D)}{\Sigma_i c(w_i; C_D)} \tag{4.7}$$

Definition 4.12 ***Quotation Likelihood Given Document***
The ***quotation likelihood*** *of* $\mathcal{Q}$ *with respect to* $\mathcal{D}$ *denoted as* $\text{score}(\mathcal{D}) = P(\mathcal{Q}|\mathcal{D})$ *is the probability that* $\mathcal{D}$ *generates quotation* $\mathcal{Q}$*. The* ***quotation likelihood*** *is estimated as:*

$$P(\mathcal{Q}|\mathcal{D}) = \prod_{w_i \in \mathcal{Q}^T} P(w_i|\mathcal{D})^{c(w_i;\mathcal{Q}^T)} \tag{4.8}$$

where $\mathcal{Q}^T$ *is the set of topic terms in* $\mathcal{Q}$*, and* $c(w_i; \mathcal{Q}^T)$ *is the count of word* w_i *in* $\mathcal{Q}^T$*.*

The set of topics for $\mathcal{Q}$ *is now* $\mathcal{Q}^{\mathcal{D}} = \{\mathcal{D}|\mathrm{score}(\mathcal{D}) > \sigma\}$*, where* σ *is a threshold (in our experiments,* $\sigma = 0.01$*, which resulted in an average number of three topics per quotation).*

4.3.5 Polarity Classification of Quotations

In the previous section, we use topic features to map a quotation onto one or more debates. In this section we describe our approach to infer the pro/con polarity of a quotation for each of the identified debates. We use the joint topic-sentiment unary and binary features, preferably but optionally with expansion. Formally, once we have a set of topics $\mathcal{Q}^{\mathcal{D}}$ *for the given quotation* $\mathcal{Q}$*, our task is to classify the polarity of* $\mathcal{Q}$ *on each* $\mathcal{D} \in \mathcal{Q}^{\mathcal{D}}$*.*

Definition 4.13 ***Pro/Con Document***
For every debate $\mathcal{D}$ *in Debatepedia, there is a set of pro documents and a set of con document. For a debate* $\mathcal{D} \in \mathcal{Q}^{\mathcal{D}}$*, we define* ***the pro document*** $\mathcal{D}^+$ *as the concatenation of* all *pro documents for that debate. Analogously, we define* ***the con document*** $\mathcal{D}^-$ *as the concatenation of all con documents for the debate.*

Our classification model of quotations into pro and con is based on the query likehood model. Given a pro and a con debate for $\mathcal{D}$*, our approach compute the probabilities of generating the quotation. The intuition is that a pro quotation is more likely to be generated by a pro document than a con document since they share more similar expressions, while a con quotation is more likely to be generated by con document than a pro document. Therefore, we compute two different quotation likelihoods:* $P(\mathcal{Q}|\mathcal{D}^+)$ *and* $P(\mathcal{Q}|\mathcal{D}^-)$*.*

Definition 4.14 ***Quotation Likelihood Given Pro/Con Document***
We estimate ***the quotation likelihood*** *with respect to both* $\mathcal{D}^+$ *as well as* $\mathcal{D}^-$ *as follows.*

$$P(\mathcal{Q}|\mathcal{D}^+) = \prod_{w_i \in \mathcal{Q}} P(w_i|\mathcal{D}^+)^{c(w_i;\mathcal{Q})} \tag{4.9}$$

$$P(\mathcal{Q}|\mathcal{D}^-) = \prod_{w_i \in \mathcal{Q}} P(w_i|\mathcal{D}^-)^{c(w_i;\mathcal{Q})} \tag{4.10}$$

If $(P(\mathcal{Q}|\mathcal{D}^+) > P(\mathcal{Q}|\mathcal{D}^-))$*, the quotation is classified as pro, otherwise, we classify it as con. In effect, this is a* k*NN classifier (k nearest neighbors) with* $k = 1$ *in our 2-class setting.*

The query is represented as a set of features where each feature is denoted as w_i*. While we can use the terms in the quotation and debates as is, this is unlikely to give us good results (as we show in our experiments) because of the sparsity of terms in the quotation. To overcome this problem, we make use of the features introduced in Sections 4.3.1 and 4.3.2 for the estimation of the language models of the debates. An overview is given in Table 4.4. We describe these features in details in the rest of this section.*

	LM and Features
LM-NG	LM over n-grams features, where $n \leq 3$
LM-UNA	LM over unary features
LM-BIN-I	LM over unary and binary features assuming independence
LM-BIN-D	LM over unary and binary features assuming dependence

Table 4.4: Overview of the different LMs

4.3.5.1 LM over n-grams

Definition 4.15 ***Debate*** n***-grams Language Model***
For a given debate $\mathcal{D}$*,* ***the debate*** n***-grams language model*** *denoted as* **LM-NG***, is estimated for each* $\mathcal{D}^+$ *and* $\mathcal{D}^-$ *over all their possible* n*-grams (e.g., where* $n \leq 3$*) in their corresponding documents as follows:*

$$P_{\mathcal{D}^+}(w) = (1 - \lambda)P(w|\mathcal{D}^+) + \lambda P(w|\mathcal{C}_D) \tag{4.11}$$

where w *is an* n*-gram and* $\mathcal{C}_D$ *is the background corpus consisting of all debates in the same branch of Debatepedia. The parameters of the language model of the debate are estimated using the maximum-likelihood estimator as follows.*

$$P(w|\mathcal{D}^+) = \frac{c(w;\mathcal{D}^+)}{\Sigma_i c(w_i;\mathcal{D}^+)} \tag{4.12}$$

$$P(w|C_D) = \frac{c(w;C_D)}{\Sigma_i c(w_i;C_D)} \tag{4.13}$$

Analogously, we also estimate the language model for $\mathcal{D}^-$*.*

Finally, we estimate the quotation likelihood with respect to both $\mathcal{D}^+$ *as well as* $\mathcal{D}^-$*.*

4.3.5.2 LM over Unary Features

Definition 4.16 ***Debate Unary Language Model***
For a given debate $\mathcal{D}$, ***the debate unary language model*** *denoted as* **LM-UNA***, are estimated for $\mathcal{D}^+$ and for $\mathcal{D}^-$ as a mixture model of two LMs, one consisting of topic terms and the other consisting of sentiment terms. Recall that topic terms and sentiment terms together form the unary features. The language model of $\mathcal{D}^+$ is estimated as:*

$$P_{\mathcal{D}^+}(w) = \alpha P_{\mathcal{D}_T^+}(w) + (1-\alpha) P_{\mathcal{D}_S^+}(w) \tag{4.14}$$

where w is a unary feature, $P_{\mathcal{D}_T^+}(w)$ is the probability of w in the topic LM of $\mathcal{D}^+$, and $P_{\mathcal{D}_S^+}(w)$ is the probability of w in the sentiment LM of $\mathcal{D}^+$, and α is a parameter which determines the importance of each. The topic LM and sentiment LM are estimated as it was described before in the previous section, with the universe of terms consisting of topic terms and sentiment terms, respectively. Analogously, we also estimate the LM for the con document $\mathcal{D}^-$.

4.3.5.3 LMs with Binary and Unary Features

Recall that binary features are $\langle t, s\rangle$ pairs, where t is a topic term and s is a sentiment term and the two are in a dependency relationship with one another (determined based on dependency parsing).

Definition 4.17 ***Debate Binary–Unary Language Model with Independent Features)***
For a given debate $\mathcal{D}$, a ***binary–Unary language model with independent features*** *denoted as* **LM-BIN-I***, is estimated for each the pro document $\mathcal{D}^+$ and the con document $\mathcal{D}^-$ over both the unary and the binary features of the corresponding document. We treat all features to be independent of each other and disregard dependencies among features. We estimate the language model of $\mathcal{D}^+$ as follows:*

$$P_{\mathcal{D}^+}(w) = \beta P_{\mathcal{D}_U^+}(w) + (1-\beta) P_{\mathcal{D}_B^+}(w) \tag{4.15}$$

where w is a unary or binary feature, $P_{\mathcal{D}_U^+}$ is the unary LM of $\mathcal{D}^+$, estimated as explained in the previous section, $P_{\mathcal{D}_B^+}$ is the binary LM which is estimated in an analogous manner to the unary LM, and β is a weighting factor. The LM for $\mathcal{D}^-$ is estimated in an analogous manner.

In order to increase the accuracy of the LM, we propose the modeling of limited dependence *among features denoted as* **LM-BIN-D**. *We do this by considering a universe of terms consisting of* pairs *of features* $\langle f_i, f_j \rangle$ *where* f_i *and* f_j *could be the unary or the binary features. For example the quotation in Table 4.3 has as features pairs ⟨fence, help⟩, ⟨fence, ⟨economy, help⟩⟩, ⟨⟨fence, help⟩, ⟨economy, support⟩⟩, etc.. This is similar to modeling bigrams in the standard LM setting, but with a crucial difference. While bigrams are two* consecutive *unigrams, we cannot insist that our feature pair are consecutive. Instead, we make the assumption that the feature pair occur in the* same sentence. *That is, the frequency of a feature pair is the number of sentences in which they co-occur. With this in mind, we estimate our new LM as the interpolation of two different LMs.*

Definition 4.18 ***Debate Binary–Unary Language Model with Dependent Features***
For a given debate $\mathcal{D}$, a ***binary–unary language model with dependent features*** *denoted as* **LM-BIN-D**, *is estimated for each the pro document $\mathcal{D}^+$ and the con document $\mathcal{D}^-$ over both the unary and the binary features of the corresponding document. We consider limited dependencies among features by regarding a feature pair that occur in the* same sentence. *We estimate the language model of $\mathcal{D}^+$ as follows:*

$$P_{\mathcal{D}^+}(w) = \beta P_{\mathcal{D}^+_U}(w) + (1-\beta) P_{\mathcal{D}^+_{pair}}(w) \tag{4.16}$$

where w is now either a unary feature, or a feature pair.

We can now compute the likelihood of generating the quotation from either the pro document or the con document, in a straightforward way. However, since our binary features are confined to the scope of the same sentence, we can alternatively compute the likelihood of generating a sentence *from the two polarities' documents. As a quotation typically consists of few sentences (between 1 and 10), we can subsequently aggregate over these sentence likelihoods. This is expressed as follows.*

Definition 4.19 ***Pro Document Score***
The ***pro document score*** *is denoted as:*

$$\mathrm{score}(\mathcal{Q}, \mathcal{D}^+) = \mathrm{MAX}_{sen_i \in \mathcal{Q}}(P(sen_i|\mathcal{D}^+)) \tag{4.17}$$

where sen_i *is a sentence in the quotation,* MAX *denotes the maximum over the likelihoods of the quotation's sentences, and* $P(\mathrm{sen}_i|\mathcal{D}^+)$ *is computed as,*

$$P(\mathrm{sen}_i|\mathcal{D}^+) = \prod_{w \in \mathrm{sen}_i} P_{\mathcal{D}^+}(w) \tag{4.18}$$

Analogously, we define the ***con document score*** $\mathrm{score}(\mathcal{Q}, \mathcal{D}^-)$.

4.3.5.4 Entailment and Contradiction Model

As our final variant, we make use of the agreement *and* disagreement *features of Section 4.3.2. The intuition here is that, not only should a pro document (respectively, con) agree with the* agreement *expressions, but the con document (respectively, pro) should agree with the* disagreement *expressions.*

Definition 4.20 ***Entailment and Contradiction Model***
Given the agreement features $\mathcal{Q}^+$ *and disagreement features* $\mathcal{Q}^-$ *of a quotation* $\mathcal{Q}$*, in the* ***entailment and contradiction model (EC)****, the probability of generating* $\mathcal{Q}$ *given either the pro document* $\mathcal{D}^+$ *or the con document* $\mathcal{D}^-$ *:*

$$P(\mathcal{Q}|\mathcal{D}^+) = (1-\lambda)P(\mathcal{Q}^+|\mathcal{D}^+) + \lambda P(\mathcal{Q}^-|\mathcal{D}^-) \tag{4.19}$$

$$P(\mathcal{Q}|\mathcal{D}^-) = (1-\lambda)P(\mathcal{Q}^+|\mathcal{D}^-) + \lambda P(\mathcal{Q}^-|\mathcal{D}^+) \tag{4.20}$$

where λ *is a weight parameter to determine the importance of the agreement features of the quotation versus the disagreement features. If* $\lambda = 0$ *which means we consider only the agreement features, we denote the model in this case as* ***E****.*

We estimate the above probabilities (e.g. $P(\mathcal{Q}|\mathcal{D}^+)$*) using the models described in the previous sections (see Table 4.4 for LM variants).*

4.4 Experimental Evaluation

We evaluated the effectiveness of the proposed features models and the quotation expansion model in combination with several classifiers: LM-based kNN (k nearest neighbors, here with the special case $k = 1$*), LDA, and SVM. We report on both the classification of quotations onto fine-grained topics, and the classification into pro/con polarities. The metrics of interest are precision and recall, both micro-averaged over all quotations in* *a test set and macro-averaged over the classes of interest (topics and pro/con, the latter being a binary case). Since there is no established benchmark for our setting, we created our own training and test datasets as explained next.*

4.4.1 Experimental Setup

4.4.1.1 Quotations Datasets

From Debatepedia, we extracted a total number of 142,253 pro and con quotations. From this set of quotations, we created the following experimental sets:

- **Test dataset***: We compiled, by random sampling, a held-out set of 250 pro and 250 con quotations from various different topics as a* test set*. These 500 quotations, each belonging to one or more topics, covered a total of 73 different fine-grained topics from Debatepedia. Since the topics as well as the polarities are given in advance, the ground truth for the classifiers is known.*
- **Development dataset***: We performed hyper-parameter tuning for all tested classification methods (see Section 4.4.1.3) on separate* development set *of 200 pro and con quotations sampled from Debatepedia for 73 topics (the topics that occur in the test dataset).*
- **Training dataset***: For training supervised classifiers, we sampled 4,400 quotations from Debatepedia for 73 topics (the topics that occur in the test dataset) using all quotations that do not belong to the development set or the test set.*

4.4.1.2 Topic Documents Variants

As previously mentioned, Debatepedia provides us with pro and con debates (documents) for many topics. We tried the following variations of preparing pro and con documents.

- **DNone** *with original features only: Only the unary and binary features extracted from the debate documents are used as features to represent the debates.*
- **DExp** *with expansion: In addition to the unary and binary features, their synonyms and antonyms are added to the features set. Therefore, the final set of features for a debate includes its unary and binary features and their expansions with synonyms and antonyms.*

4.4.1.3 Methods under Comparison

We compared our family of LM-based methods against each other and against two baselines.

Baselines *In Sections 4.3.1 and 4.3.2, we defined a variety of features. Instead of using these to estimate LMs, we can directly use them as features for other classification algorithms.*

- *LDA (Latent Dirichlet Allocation): a state-of-the-art latent-topic clustering method [113, 21].*
- *SVM (Support Vector Machine): a supervised discriminative classifier [60].*

Equipped with various feature models (n-grams, unary with expansions and binary with expansions), both LDA and SVM were trained with the quotations in the training dataset described in Section 4.4.1.1, and tuned with the separate development set.

LM-based classification *We studied the LM-based methods described in Section 4.3.5 on different test sets, using different feature models, and tuned with the separate development set:*

- *The n-grams model, denoted as **LM-NG**.*
- *The entailment model given the pro LM and the con LM, in combination with:*
  1. *the unary model only denoted as **LM-E-UNA**,*
  2. *the binary and the unary models assuming binary features independence, denoted as **LM-E-BIN-I**,*
  3. *the binary and the unary models assuming binary features dependence, denoted as **LM-E-BIN-D**.*
- *The entailment and contradiction model given the pro LM and the con LM, in combination with:*
  1. *the unary model only, denoted as **LM-EC-UNA**,*
  2. *the binary and the unary models assuming binary features independence, denoted as **LM-EC-BIN-I**,*
  3. *the binary and the unary models assuming binary features dependence, denoted as **LM-EC-BIN-D**.*

4.4.2 Classification Methods based on LM

We conducted experiments with our family of LM-based methods on mapping quotations to topics and on classifying them into pro/con polarities (for each topic). We used two datasets in these experiments:

1. *The* **Debatepedia test dataset** *described in Section 4.4.1.1.*

2. **ProCon test dataset***:* `www.procon.org` *is a political website which provides pro and con quotations by politicians on specific topics. Each quotation is tagged by both the topic and the stance (pro or con). We collected 400 quotations on various topics as our second test set.*

   Note that the topics in ProCon are not the same as the topics in Debatepedia. Therefore for the assessment of mapping quotations to topics we relied on manual assessments by two judges.

   Note also that the stance (pro or con) of a quotation on a topic in ProCon may differ from the stance in Debatepedia (e.g., pro "immigration amnesty" could become con "deporting illegal immigrants"). After the manual assessments of the 400 quotations mapping to topics, we found that 235 quotations have their topics in the 73 topics of the test dataset collected from Debatepedia. Therefore, we have manually created a ground-truth set of pro and con labels for these 235 quotations.

4.4.2.1 Mapping Quotations to Topics

Debatepedia dataset *A quotation in Debatepedia belongs to one or more debates. We denote the set of debates that a quotation Q belongs to as $Q^{\mathcal{D}}$. We computed the recall and precision of mapping 500 test quotations $|Q| = 500$ onto debates as follows.*

Let TP *be the number of debates to which* Q *is* correctly *mapped. Let* FP *be the number of debates to which* Q *is* incorrectly *mapped. Let* FN *be the number of debates to which* Q *should have been mapped, but are missed by the approach. The* precision *is computed for each quotation as:*

$$\mathsf{Precision} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}} \tag{4.21}$$

while the recall *is computed as:*

$$\mathsf{Recall} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}} \tag{4.22}$$

Macro-averaged *scores are calculated by first calculating precision and recall for each quotation and then taking the average of these.*

$$\mathrm{Precision}_{\mathrm{Macro}} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{TP_i}{TP_i + FP_i} \tag{4.23}$$

$$\mathrm{Recall}_{\mathrm{Macro}} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{TP_i}{TP_i + FN_i} \tag{4.24}$$

Micro-averaged *scores are calculated by summing up the individual precisions and recalls.*

$$\mathrm{Precision}_{\mathrm{Micro}} = \frac{\sum_{i=1}^{|Q|} TP_i}{\sum_{i=1}^{|Q|} TP_i + FP_i} \tag{4.25}$$

$$\mathrm{Recall}_{\mathrm{Micro}} = \frac{\sum_{i=1}^{|Q|} TP_i}{\sum_{i=1}^{|Q|} TP_i + FN_i} \tag{4.26}$$

We were conservative in our assessments. We considered a mapping by our methods as correct only if we exactly matched the ground-truth debate, thus discounting "near misses" (e.g., mapping onto "Gay rights" debate when the ground-truth debate is "LGBT adoption").

The micro- and macro-averaged precision and the micro- and macro-averaged recall for this experiment are given in Table 4.5.

#Quotations	Micro Precision	Micro Recall	Macro Precision	Macro Recall
500	**0.72**	**0.81**	**0.78**	**0.85**

Table 4.5: Topic classification of Debatepedia dataset

ProCon dataset *Recall that we created a random sample of 400 quotations, with mappings to Debatepedia topics manually assessed by two judges.*

For assessing the reliability of agreement between the two judges, we use the Cohen's kappa coefficient as a statistical measure [31]. In our case each judge classifies 400 quotations items into 2 mutually exclusive categories "correct" and "wrong". The equation for the Kappa coefficient is:

$$\kappa = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)} \tag{4.27}$$

$\Pr(a)$ *is the relative observed agreement among judges, and* $\Pr(e)$ *is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each judge randomly saying each category. If the judges are in complete agreement then* $\kappa = 1$. *If there is no agreement among the judges other than what would be expected by chance (as defined by* $\Pr(e)$*), then* $\kappa = 0$.

Table 4.6 shows the results for the micro- and macro-averaged precision of mapping quotations to topics, and also Cohen's kappa for the agreement between the two judges. In this setting, there is no way of measuring recall since we do not have a ground-truth set of topics for the quotations in this dataset.

	Judge 1	Judge 2
#correct	337	351
#wrong	63	49
Micro Precision	0.84	0.88
Macro Precision	0.87	0.90
kappa coefficient	**0.84**	

Table 4.6: Topic classification of ProCon dataset

4.4.2.2 Pro/Con Classification

Debatepedia dataset *Results of the LM-based approaches with the different topic documents variants are shown in Table 4.7. These are the results found at* $\alpha = 0.64$ *for the LMs over the unary features (**LM-UNA**),* $\beta = 0.18$ *for the LMs over the unary and the binary features (**LM-BIN**), and* $\lambda = 0.24$ *for the entailment and contradiction model (**EC**). The hyper-parameter values are automatically determined from the development set.*

At test level $\alpha = 0.05$ *using the paired two sample t-test, we found that the results of the different techniques on **DNone** and **DExp** are not statistically different. This means that the expansion of the debates did not significantly improve the results. On the other hand, the difference in the results between the **E** models which use only the agreement features of the quotations and the **EC** models which use both the agreement and the disagreement features is statistically significant. So quotation expansion using both the synonyms and the antonyms improves the results. In addition, the difference in the results of the **BIN-I** model, and the **BIN-D** model were statistically significant,*

	DNone	DExp
LM-NG	0.68/0.73	0.68/0.75
LM-E-UNA	0.68/0.66	0.69/0.64
LM-E-BIN-I	0.71/0.68	0.70/0.65
LM-E-BIN-D	0.69/0.75	0.72/0.76
LM-EC-UNA	0.65/0.67	0.66/0.69
LM-EC-BIN-I	0.70/0.76	0.72/0.75
LM-EC-BIN-D	0.73/0.76	**0.74/0.78**

Table 4.7: Micro-averaged precision/recall for LM-based pro/con classification on Debatepedia test set

too, which means that considering the dependency of the binary features in each sentence improves the results.

ProCon dataset *We evaluated a set of 235 quotations from the set of quotations from the ProCon test set assigned to the topics in Debatepedia test set. We considered the classification models (**LM-EC-BIN-D**, **LM-EC-UNA**, and **LM-NG**). For this experiment, we used the hyper-parameter values of the LM determined from the development set (e.g.* $\alpha = 0.64$, $\beta = 0.18$, *and* $\lambda = 0.24$). *Table 4.8 shows the results of the three different models. These results are statistically significant at test level* $\alpha = 0.05$ *using the paired two sample t-test.*

	Micro Precision	Micro Recall	Macro Precision	Macro Recall
LM-NG	0.67	0.68	0.70	0.69
LM-EC-UNA	0.64	0.67	0.65	0.69
LM-EC-BIN-D	**0.72**	**0.70**	**0.71**	**0.71**

Table 4.8: Micro- and macro-averaged precision/recall for LM-based pro/con classifications on the ProCon test set

4.4.3 Pro/Con Classification with LDA & SVM

In Sections 4.3.1 and 4.3.2, we defined a variety of features. Instead of using these to estimate LMs, we can directly use them as features in a clustering or a classification al-

gorithm. We require that the quotations of one topic be classified into exactly 2 clusters ($K = 2$), a pro cluster and a con cluster. Therefore, we conducted experiments on the Debatepedia test set in order to evaluate the effectiveness of the proposed feature models and quotation expansion model on the pro/con classification task with two different classifiers SVM and LDA, in comparison to our LM-based methods. With our patterns (n-grams, unary and binary with expansion) as features, both LDA and SVM were trained with quotations in Debatepedia that belong to the 73 topics that the held-out Debatepedia dataset belongs to. In total 4,400 quotations used to train the classifiers, with average number of 60 quotations per topic. The two classifiers are trained for each Debatepedia topic separately. The held-out dataset of Debatepedia, is used to test the classifiers.

4.4.3.1 LDA

Topic models provide a way to analyze large volumes of unlabeled text. A "topic" consists of a cluster of words that frequently occur together. Using contextual clues, topic models can connect words with similar meanings and distinguish between uses of words with multiple meanings. We used the Mallet software package[2]) which includes implementation of Gibbs sampling , methods for document-topic hyperparameter optimization, and tools for inferring topics for new documents given trained models. For pro/con classification, LDA is configured with two latent dimensions. It is trained separately for each Debatepedia topic, where the number of topics of the LDA algorithm is set to two. The number of sampling iterations is set to 2000 while the optimization interval is every 10 iterations. We then created a topic inference tools based on the trained models.

4.4.3.2 SVM

SVM is a supervised discriminative classifier (implemented using the SVMlight software package[3]). For each Debatepedia topic, we train a binary classifier with a linear kernel. The performance of the linear SVM classifier depends on the choice of the regularization parameter C. To choose C using 5-fold Cross-validation (CV), we first split the availbale data for each Debatepedia topic into 5 subsets. Then we compute the CV error using this split error for the SVM classifiers using different values for C. Finally,

[2] `mallet.cs.umass.edu`
[3] `svmlight.joachims.org`

we pick the C with the lowest CV error and use it for training an SVM on the complete data set of a topic.

4.4.3.3 Results

	BIN-D with expansion			
	Micro Precision	Micro Recall	Macro Precision	Macro Recall
LDA	0.72	0.76	0.70	0.74
SVM	0.71	0.72	0.71	0.78
LM	**0.74**	**0.78**	**0.70**	**0.80**
	UNA with expansion			
	Micro Precision	Micro Recall	Macro Precision	Macro Recall
LDA	0.63	0.70	0.67	0.77
SVM	0.60	0.68	0.65	0.72
LM	0.66	0.69	0.71	0.76
	NG			
	Micro Precision	Micro Recall	Macro Precision	Macro Recall
LDA	0.67	0.77	0.69	0.81
SVM	0.63	0.74	0.67	0.81
LM	0.68	0.75	0.73	0.80

Table 4.9: LM pro/con classifiers micro- and macro-averaged precision and recall compared to SVM and LDA

Table 4.9 shows the micro- and macro-averaged precision and the micro- and macro-averaged recall of LDA and SVM compared to the LM-based approaches in combination with different feature models (e.g. ***LM-EC-BIN-D, LM-EC-UNA, and LM-NG****). The results are statistically significant at test level* $\alpha = 0.05$ *using the paired two sample t-test. We notice that the binary features model with expansion improved the results of both the SVM and the LDA classifiers. Moreover, our LM-based methods outperformed SVM and LDA by a significant margin for both precision and recall.*

4.4.4 Political Bias of News Media

We further demonstrate the usefulness of our methods for applications like analyzing and visualizing the political bias of different news media. For this study, we used two groups of quotations:

1. *A group of six sets of quotations collected from news articles by formulating Google queries for three controversial topics (gun control, gay rights, and capital punishment) and restricted to two media sources (***CNN and Fox News***). A corpus is created for each pair of topic and media source, each with its top 50 Google results. This way, we obtained 30 quotations for each topic from each media source.*

   In total, this gave us a test collection of 180 quotations. The quotations were automatically extracted from the news articles using phrases like ("said", "told", etc.). We excluded the quotations with $\texttt{score}(\mathcal{D}) \leq 0.01$ *(as described in Section 4.3.4) for their top debates. Three judges assessed the pro/con polarities for all 180 quotaions.*

2. *A group of quotations collected from the newspaper websites of Aljazeera, Jerusalem Post, New York Times, and Der Spiegel on two topics:* **WikiLeaks and Occupy Protests***. Each newspaper source contributed 30 quotations (collected as described before by querying Google), for a total of 240 quotations. Again, we obtained manual assessments by three judges.*

The quotations' topics in these groups belong to the 73 topics of the Debatepedia test dataset, and therefore we used the same hyper-parameter values determined from the development set of Debatepedia.

For the CNN and Fox News quotations group, Table 4.10 shows the micro-averaged precision and recall of the quotations pro/con classification for the topic and media pairs. The compared methods are the LM approach with the best results ***LM-EC-BIN-D*** *and a classification method based on the party of the opinion holders. The party-based assessment assumes [4] that democrats are pro gun control, pro gay rights, and con capital punishment. On the other hand, republicans are con gun control and gay rights, but pro capital punishment. We annotated each extracted quotation by the party of its opinion holder (politician) using opencongress database (www.opencongress.org). The results show that the party-based classification of quotations is not reliable.*

[4] `www.diffen.com`

	#Evaluated Quotations	LM Pro/Con Classification	**Party-based** Classification
Fox News:			
Gun Control	30	0.76/0.87	0.60/0.60
Gay Rights	30	0.67/0.67	0.60/0.40
Capital Punishment	30	0.76/0.87	0.56/0.60
CNN:			
Gun Control	30	0.70/0.80	0.73/0.53
Gay Rights	30	0.73/0.73	0.63/0.33
Capital Punishment	30	0.70/0.93	0.67/0.80
Micro Precision/Recall	180	**0.72/0.80**	**0.63/0.54**

Table 4.10: CNN & Fox News micro-averaged precision and recall

For the WikiLeaks and Occupy protests quotations group, Table 4.11 shows the micro-averaged precision and recall of the quotations pro/con classification for the topic and media pairs. Again, the used pro/con classification method is the ***LM-EC-BIN-D****.*

	#Evaluated Quotations	LM Pro/Con Classification
Aljazeera:		
WikiLeaks	30	0.73/0.73
Occupy Protests	30	0.67/0.67
Jerusalem Post:		
WikiLeaks	30	0.70/0.80
Occupy Protests	30	0.76/0.86
New York Times:		
WikiLeaks	30	0.63/0.67
Occupy Protests	30	0.73/0.53
Spiegel:		
WikiLeaks	30	0.77/0.67
Occupy Protests	30	0.70/0.80
Micro Precision/Recall	240	**0.71/0.72**

Table 4.11: WikiLeaks & Occupy Protests micro-averaged precision and recall

In Table 4.12 we list some of the extracted quotations and their classification into pro and con from the two groups of quotations.

4.4.5 Discussion

For the classification onto topics, our precision results of about 80 to 85 % are amazingly good, given that we map quotations onto a large set of fine-grained debates.

For the pro/con assignment, our best method **LM-EC-BIN-D** *achieves almost 74% precision. It uses the richest features, the dependent pairs of binary features and the entailment-contradiction expansions. While one may have hoped for even higher precision, this is actually a decent result given the sophisticated nature and stylistic subtleties of political quotations. The gains over the simpler alternatives are statistically significant. This shows that the novel elements in our features model and quotation expansion are indeed decisive for achieving good precision on this difficult classification task. The experimental results also show that our features model are decisive for achieving good pro/con classification precision using classification methods such as SVM and LDA. The overall winner, however, in this comparison is the LM-based method with rich features and quotation expansion.*

The usefulness of our methods is further demonstrated by enabling automatic visualization of entire quotations corpora. The heatmap of topic-media pairs shown in Figure 1 was actually generated by our software, in a fully automated manner.

4.5 Related Work

Many previous works on polarity analysis address the problem of topic independent *opinion mining. For example, classifying a movie review into positive or negative. However, our setting is* topic dependent *and the classification of a document into pro or con can change depending on the topic (see [39]).*

On the otherhand, many previous methods rely on training classifiers with annotated training data (see, e.g., [94, 119]); manual annotation is often a bottleneck. In our work, we follow an alternative approach of using language models (LMs) to classify opinions, thus reducing the dependence on annotated training data.

Other studies of polarity analysis focus on detecting text polarity given that the classified documents are part of social media like online debates, blogs and twitters

Fox News:	
Gun Control **(pro)**:	**(10-Jan-2011)** "Guns kill. And those who glamorize gunplay or worship gun ownership do no service to humanity." **(Carolyn Maloney, D-N.Y.)**
Gay Rights **(con)**:	**(07-Feb-2012)** "homosexuality poses a serious threat to family. The bill has helped raise public awareness about the dangers to our children" **(David Bahati)**
Capital Punish. **(pro)**:	**(27-March-2010)** "The death penalty provides a sense of justice to the system, is a just punishment for murder and has a deterrent effect on crime" **(Paul Cassell)**
CNN:	
Gun Control **(con)**:	**(12-May-2011)** "I'm against gun control for the reason, it doesn't affect the bad guys, because they're going to have guns" (**Donald Trump)**
Gay Rights **(pro)**:	**(10-Oct-2011)** "We applaud the administration's progress, while we also encourage him to 'evolve faster' on supporting full marriage equality" **(Stuart Gaffney)**
Capital Punish. **(con)**:	**(28-March-2011)** "Any country that continues to execute is flying in the face of the fact that both human rights law and UN human rights bodies consistently hold that abolition should be the objective" **(Salil Shetty)**
Aljazeera:	
WikiLeaks **(pro)**:	**(01-Sep-2011)** "the cables' release also played a role in setting off the mass movement that has jolted dictatorial regimes across the Arab world" **(WikiLeaks)**
Occupy Protest **(pro)**:	**(04-Oct-2011)** "Occupy Wall Street is not only a political protest, but it's also a model society, which I think is the really interesting political protest– that it is itself the demand" **(Jesse A. Meyerson)**
Jerusalem Post:	
WikiLeaks **(con)**:	**(27-Nov-2010)** "We are all bracing for what may be coming and condemn WikiLeaks for the release of classified material. It will place lives and interests at risk. It is irresponsible." **(State Department spokesman)**
Occupy Protest **(pro)**:	**(16-Nov-2011)** "Occupy Judaism stands shoulder to shoulder with the Occupy Wall Street protesters" **(Jewish protesters)**
New York Times:	
WikiLeaks **(con)**:	**(28-Nov-2010)** "We condemn in the strongest terms the unauthorized disclosure of classified documents and sensitive national security information" **(White House)**
Occupy Protest **(con)**:	**(4-Feb-2012)** "The Bay Area Occupy Movement has got to stop using Oakland as their playground" **(Mayor Quan)**
Spiegel:	
WikiLeaks **(pro)**:	**(2 Sep 2011)** "Shining a light on 45 years of US 'diplomacy,' it is time to open the archives forever" **(WikiLeaks)**
Occupy Protest **(pro)**:	**(10-Nov-2011)** "America had become a society divided between rich and poor, in which the poor and working classes are squeezed" **(Time Magazine)**

Table 4.12: Examples for extracted and classified quotations

[110, 112, 83, 32, 26, 1]. These studies rely on the polarities of the features of the text, which are rich in these types of media. Some of these studies further consider the linkage among documents to detect polarities [1, 119, 26]. However polarized features and hyperlinks are very sparse in news media.

4.5.1 Wordnet and Polarity Analysis

Many approaches to polarity analysis rely on lexicons of words that may be used to express subjectivity. Most subjectivity lexicons are compiled as lists of keywords, rather than word meanings (senses). However, many keywords have both subjective and objective senses. Subjectivity clues used with objective senses are a significant source of error in subjectivity and sentiment analysis. To tackle this source of error, many works proposed using WordNet for word sense disambiguation.

[13] is an approach that uses semi-supervised machine learning methods to determine polarity of subjective terms by exploiting information given in glosses provided by WordNet. In particular this approach is based on the assumption that terms with similar orientation tend to have similar glosses. Therefore, by means of glosses classification authors aim to classify the terms described by these glosses.

[4] defines a subjectivity word sense disambiguation task, which is to automatically determine which word instances in a corpus are being used with subjective senses, and which are being used with objective senses. Their system relies on common machine learning features for word sense disambiguation. They report performance of rule-based polarity classification using word senses. The used classifier labels instances of lexicon entries in two-step approach. The first step classifies keyword instances as being in a polar (positive or negative) or a neutral context. The first step is performed by the neutral/polar classifier. The second step decides the contextual polarity (positive or negative) of the instances classified as polar in the first step, and is performed by a separate classifier.

The work presented in [105] suggests using word senses to detect sentence level polarity of news headlines. The authors use graph similarity to detect polarity of senses. To predict sentence level polarity, a HMM is trained on word sense and POS as the observation. The authors report that word senses particularly help understanding metaphors in these sentences.

In [35], the authors suggest expansion using WordNet relations. Their system is based on the identification of concepts in the sentences rather than terms, using a word *sense disambiguation tool to obtain the correct senses for these concepts. They construct feature vectors that map to a larger sense-based space. In order to do so, they use synset offsets as representation of sense-based features. The WordNet Affect lexicon is used to identify those concepts which are a priori candidates to denote an emotion or feeling.*

The work in [48] discusses algorithmic construction of sentiment dictionaries. Their method expands small candidate seed lists of positive and negative words into full sentiment lexicons using path-based analysis of synonym and antonym sets in WordNet. They use sentiment-alternation hop counts to determine the polarity strength of the candidate terms and eliminate the ambiguous terms. They associate a polarity (positive or negative) to each word, and query both the synonyms and antonyms. Synonyms inherit the polarity from the parent, whereas antonyms get the opposite polarity. The significance of a path decreases as a function of its length or depth from a seed word.

The work described in [104] explores incorporation of semantics in a supervised sentiment classifier. They use the synsets in Wordnet as the feature space to represent word senses. Thus, a document consisting of words gets mapped to a document consisting of corresponding word senses. The use of WordNet similarity metrics helps in mitigating unknown synsets in the test corpus by replacing them with known synsets in the training corpus. They conducted sets of experiments to highlight their hypothesis that WordNet senses are better features as compared to words. Their synset replacement algorithm uses Wordnet similarity-based metrics which replace an unknown synset in the test corpus with the closest approximation in the training corpus.

4.5.2 Topic-Polarity Statistical Models

Prior work addressed the problem of detecting general perspectives (e.g. ideologies or political parties) of given texts, e.g., Republicans versus Democrates, or Palestinian versus Israeli [119, 86, 133, 79, 138, 91]. These studies use statistical methods and train classifiers on a set of perspective-annotated documents in order to learn a set of discriminative n*-grams of each perspective. These methods require manually annotated documents, which is not practical if we move to a finer level of granularity.*

Coarse-grained classification is taken further by the work of [62], which aims to annotate political speeches and parliament debates, but does not deal with fine-grained topics. Other studies use prior knowledge about the parties or the ideologies of the opinion holders mentioned in the documents [80]. For example, when an article has many quoted phrases by Barack Obama or Hillary Clinton, then most probably this article is written *from a democratic perspective. However this prior knowledge about the opinion holders' ideologies or parties is not always available, and often fairly crude.*

[84] uses semantic taxonomies to identify facets of topics, and analyzes opinions on these facets rather than topics as a whole. This applies to opinion mining on politicians (with facets such as Vietnam war, Watergate affair, etc.), but it does not address the polarity issue of these opinions. [45] examines opinions at the collection level with each collection on a topic coming from a different perspective. A latent topic model is devised to discover the common topics across all the perspectives. For each topic, the opinions from each perspective are summarized. A related task is addressed in [17]. They focus on predicting the polarity of comments on blog postings. Their approach models mixed-community responses, to identify the topics and the responses they evoke in different sub-communities.

Language Models have been applied to various opinion-mining and polarity-analysis tasks [93, 79, 82, 34, 90, 66, 78]. For example, [79] ranks sentences by both sentiment relevancy and topic relevancy. The described work proposes a generative model that jointly models sentiment words, topic words, and sentiment polarity in a sentence as a triple. [186] examines the difference of two collections of different perspectives using the Kullback-Leibler (KL) divergence between posterior distributions induced from the document collection pairs, and discover that the KL divergence between different aspects is an order of magnitude smaller than that between different topics.

[82] utilizes manually labeled data and noisy labeled data and integrates these two different kinds of data into the same learning framework. The basic idea is to train a language model based on the manually labeled data, and then use the noisy emoticon data for smoothing.

The approach described in [90] uses statistical modeling to model review comments. Two generative models are proposed to simultaneously discover and model topics and different types of comment expressions (e.g., thumbs-up , thumbs-down , question, answer acknowledgement , disagreement, agreement). The first model separates topics and comment expressions types using a switch variable and treats posts as random mixtures over latent topics and comment expressions types. The second model improves the first model by using Maximum-Entropy priors to guide topic/expression switching.

Other research works employ probabilistic latent semantic analysis or latent Dirichlet allocation to infer language models that correspond to unobserved "factors" in the data, with the hope that the factors that are learned represent topics or sentiment categories [87, 120, 121, 85, 124, 109, 75, 45].

4.6 Summary

We addressed the problem of automatically classifying indirect opinions about political debates which appear in news media and online forums in different forms (e.g. quotations by politicians or other opinion makers), into fine-grained controversial topics and a pro/con polarity for each topic. We proposed a topic/polarity classification *approach that maps indirect opinions onto one or more topics in a category system of political debates, containing more than a thousand fine-grained topics. Our method builds on the estimation of statistical language models on a variety of advanced features. These features were specifically designed to overcome a major hurdle: the brevity of quotations leading to sparseness of features. We showed the effectiveness of our techniques through systematic experiments on more than 1000 quotations gathered from the Web on a variety of topics. Our best method achieved a precision of about 74%, quite a positive result considering the hardness of the problem.*

Chapter 5

Topics Organization

5.1 Introduction

*In Chapters 3 and 4, we presented our approaches of extracting opinions from online news sources on the Web. The outcome of these approaches is a crisp set of structured records (*quadruples*) of the form:* ⟨`opinion holder`⟩ ⟨`polarity`⟩ ⟨`opinion target`⟩⟨context⟩*. The opinion targets represent fine-grained topics derived from Debatepedia. For convenient exploration and knowledge discovery at different granularities, we describe in this chapter our approach of imposing a* topic hierarchy *on the fine-grained topics (see Figure 5.1 for an overview). The fine-grained topics are derived from Debatepedia. Since Wikipedia is much richer but controversial topics are only a small part of it, we use Debatepedia as a bootstrapping asset to focus on controversies, but eventually map the extracted facets and quotations to Wikipedia categories for a richer organization of topics. Figure 5.2 shows an example of what the final hierarchy looks like in the context of the recent Arab Spring. Some of the raw facets we acquired are shown in red boxes (bottom level of the hierarchy) and include a variety of facets: "U.N. Mandate against Libya", "use of force against civilians in Libya", etc.. These raw facets are then canonicalized to "2011 military intervention in Libya" and "Gaddaffi's response to the 2011 Libyan Civil War" respectively which in turn are part of the larger topic "2011 Libyan Civil War".*

In the rest of this chapter, we first describe our approach of acquiring political debates in Section 5.2. The method we follow for the canonicalization of facets is described in Section 5.3. Section 5.4 outlines our approach of imposing a topic hierarchy on the fine-grained topics. Section 5.5 presents our experimental evaluation. Section 5.6 discusses related work. We conclude with Section 5.7.

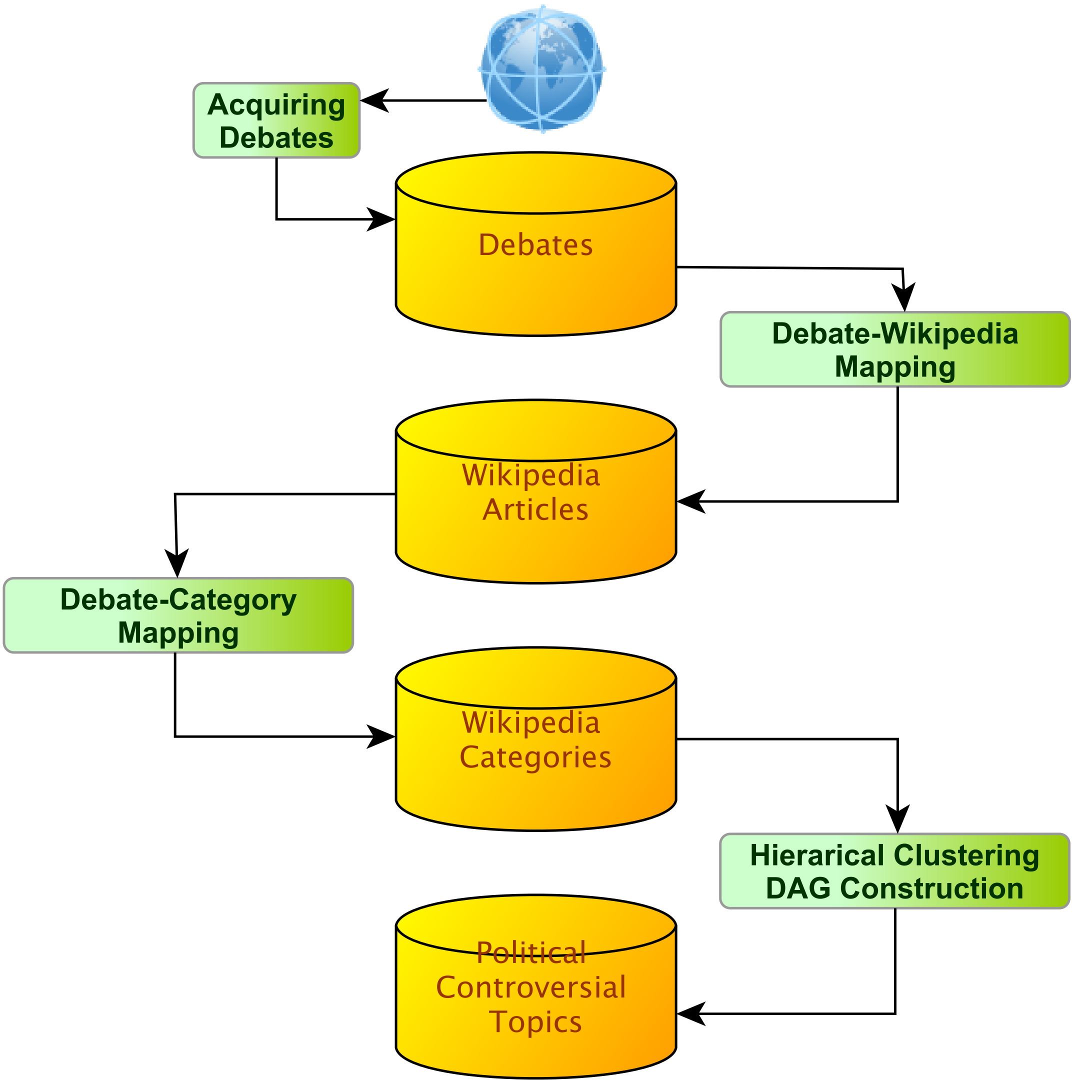


Figure 5.1: Organizing topics

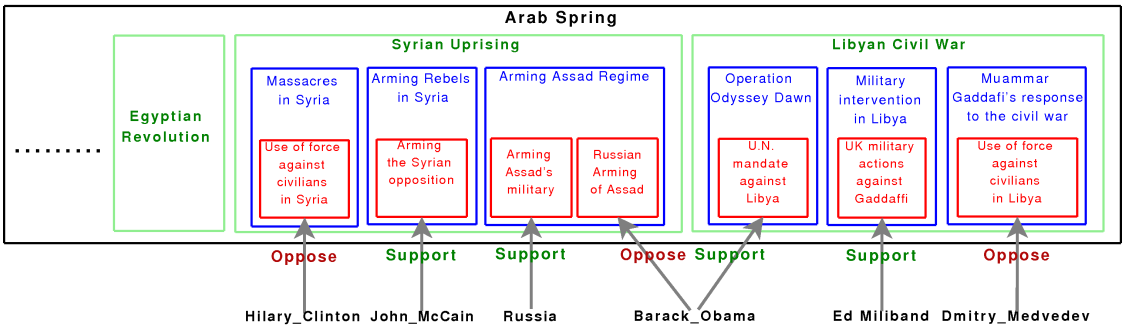


Figure 5.2: Organizing topics in different levels of granularities

5.2 Acquiring Political Debates

"opposition to", "criticisms of", "opponents argue that", "opposers argue that", "proponents claim that", "supporters claim that", "opponents claim that", "opposers claim that", "arguments for", "arguments against", "arguments in support", "arguments in favor", "arguments in opposition"

Table 5.1: Set of patterns used to collect debates from Wikipedia

As an initial step towards increasing the number of controversial fine-grained topics, we automatically build a number of debates from Wikipedia articles. To do this, we use a set of patterns in order to extract debating arguments. The existence of such patterns in a Wikipedia article page, most probably indicates that the topic of that article is of a controversial nature. We extract the sentences or the the paragraphs from the article page where these patterns occur as candidate arguments. Further more, notice that the patterns also indicate the polarity of the associated arguments (e.g. pro or con the topic). Table 5.1 lists the patterns we used to collect pro and con arguments for controversial topics from Wikipedia. Table 5.2 gives some examples of debates extracted from Wikipedia. The set of collected debates from Debatepedia or those built automatically from Wikipedia form the set of fine-grained topics that the facets and the quotations should be mapped to.

SOPA/PIPA	
Pro	Con
They are needed to protect the intellectual property of owners of content	They endanger free speech and free expression by harmfully regulating the internet
Web mining	
Pro	Con
This technology has enabled e-commerce to do personalized marketing , which eventually results in higher trade volumes. Companies can understand the needs of the customer better and they can react to customer needs faster.	The most criticized ethical issue involving web usage mining is the invasion of privacy. Another important concern is that the companies collecting the data for a specific purpose might use the data for a totally different purpose , and this essentially violates the user's interests.
Future enlargement of the European Union	
Pro	Con
It is a key regional power with a large economy and the second largest military force of NATO that will enhance the EU's position as a global geostrategic player; given Turkey's geographic location and economic, political and cultural ties in regions with that are in the immediate vicinity of the EU's geopolitical sphere of influence; such as the East Mediterranean and Black Sea coasts, the Balkan peninsula, the Middle East, the Caspian Sea basin and Central Asia.	Turkey does not respect the key principles that are expected in a liberal democracy , such as the freedom of expression, with potentially repressive laws like Article 301 (A law which states it is illegal to "insult the Turkish nation"). Turkey's large population would also alter the balance of power in the representative European institutions. Upon joining the EU, Turkey's 70 million inhabitants would bestow it the second largest number of MEPs in the European Parliament. Demographic projections indicate that Turkey would surpass Germany in the number of seats by 2020.

Table 5.2: Debates along their pro and con arguments extracted from Wikipedia

5.3 Facets Canonicalization

Chapter 3 described our approach to extract direct opinions for raw facets. In order to ensure uniformity in referring to semantically equivalent facets (e.g., "financial aid for Greece" and "EU loan to Greece" are equivalent), we automatically derive fine-grained topics or canonical names. These include a variety of facets: "U.N. Mandate against Libya", "arming the Syrian opposition", etc.. These raw facets are then canonicalized to "Military action in Libya" and "arming rebels in Syria" respectively.

For canonicalizing facets, we devised a two-step mapping *technique, based on Debatepedia and Wikipedia.*

5.3.1 Mapping Facets to Debates

For canonicalizing facets, we devised a mapping technique, based on a set of fine-grained political controversial topics represented by a set of debates from Debatepedia (`debatepedia.org`).

To ensure that our facets are indeed of a controversial nature, we first build a classifier that maps facets onto debates. To this end, we build a nearest-neighbor classifier that uses statistical language models as the basis for its distance measure. We also consider alternative classifiers like Bayesian or SVM models, but the emphasis here is on the feature space and the kNN method works very well.

Definition 5.1 ***Debate Language Model***
Let $T = \{T_1, T_2, \ldots, T_m\}$ be the set of topics in Debatepedia and let D_T denote the set of all documents debating the topics in T. Recall that each topic has many different facets that may be debated. Let D_i be the set of documents which debate the various facets of the topic T_i. The language model for a debate D_i, *smoothed with all debates D_T, is the following probability distribution over words (or phrases):*

$$P_{D_i}(w) = (1 - \lambda)P(w|D_i) + \lambda P(w|D_T) \tag{5.1}$$

where w is a word, $P_{D_i}(w)$ is the estimated probability of the word in the LM of D_i, $P(w|D_i)$ is the probability of the word in D_i and $P(w|D_T)$ is the probability of w in the "background corpus" D_T, consisting of all debates in T, and λ is a Jelinek-Mercer-style smoothing coefficient (or derived from a Dirichlet smoothing model).

We now map a raw facet f onto its nearest debate D (treating f as a query in LM-based IR terminology [53, 136]): that is, the D with maximum likelihood of generating f.

Definition 5.2 ***Ranking of Debate***
The **ranking of a debate** D_i *for a given facet* f *is in descending order of:*

$$P(f|D_i) = \prod_{w_j \in f} P_{D_i}(w_j) \tag{5.2}$$

We compute the mean of the LM scores of the top-5 debates, and the mean of the LM scores of the bottom-5 debates. If the two means are statistically different using the T-test at $\alpha = 0.05$*, the top-ranked debates are chosen as candidate names for a canonical facet. Otherwise facets are discarded.*

5.3.2 Mapping Debates to Wikipedia Articles

We extend the set of debates in Debatepedia by mapping each debate to their related articles in Wikipedia, using an LM-based kNN method. For example, the recent Libyan civil war is covered in "No-fly zone over Libya", "Human rights violation in the 2011 Libyan civil war", and "National transitional council". These titles are different facets of the same topic.

For finding the best Wikipedia articles, we would like to avoid LM comparisons against a large number of articles. Therefore, we first derive a Google query from the debate D *and restrict the search results to Wikipedia. Since the entire text of* D *would be unsuitable as input to a search engine, we generate the query from the words in the title and the ten most frequent noun phrases in* D*. The top-10 search results* $A_1, A_2, \ldots, A_{10}$ *(corresponding to Wikipedia articles) are our candidates for the* D*-to-*A *mapping.*

Definition 5.3 ***Wikipedia Article Language Model***
The **language model for Wikipedia article** A*, smoothed with the entire Wikipedia as a background corpus W, is the probability distribution over words (or phrases):*

$$P_A(w) = (1-\lambda)P(w|A) + \lambda P(w|W) \tag{5.3}$$

Definition 5.4 ***Ranking of Wikipedia Articles***
The **ranking of articles** A_i *for a given debate* D *is in descending order of:*

$$P(D|A_i) = \prod_{w_j \in D} P_{A_i}(w_j) \tag{5.4}$$

Alternatively, to this LM-based ranking Wikipedia articles for the D*-to-*A *mapping, we could use simpler word-overlap measures like Jaccard coefficients. For the final* *choice, we compute the Jaccard similarity of the set of terms between the debate query* q *and each of the top-10 candidate* A *results:*

$$J(D, A) = \frac{|\text{terms}_A \cap \text{terms}_D|}{|\text{terms}_A \cup \text{terms}_D|} \tag{5.5}$$

Our experiments did not show any major differences among the various choices for the distance measure.

5.4 Constructing the Topic Hierarchy

Once all raw facets have been mapped to canonical facet names and are associated with the corresponding Wikipedia articles, the next task is to organize them into a hierarchy of topics.

5.4.1 Wikipedia Categories

A seemingly natural approach would be to use the Wikipedia category system. However, a major problem with such an approach is that not all categories to which an article belongs are really of controversial nature For example, the raw facet "U.S sanctions against Iran" is mapped to the article ⟨U.S Sanction against Iran⟩, *which in turn belongs to different categories. Among these, "History of Iran", for example, is not relevant at all for the purpose of OpinioNetIt,*

To make this important distinction, we devise the following approach. First we collect all Wikipedia categories associated with the 3 articles to which a facet F *and its corresponding debate* D *were mapped, into candidates pool* $C_w = \{c_{w_1}, c_{w_2}, \dots\}$*. Second, we use the Debatepedia categories* $C_d = \{c_{d_1}, c_{d_2}, \dots\}$ *of the debate* D *to generate Google queries, restricted to Wikipedia categories (analogous to to the technique in Section 5.3.2). For each* c_{d_i}*, we obtain top-10 Wikipedia category pages but add only the three highest ranked ones to the candidates pool* C_w*. This step serves to reduce the candidate space. Subsequently, we employ LM-based kNN mapping to obtain the Wikipedia categories from* C_w *for the debate* D*.*

Definition 5.5 ***Wikipedia Category Language Model***
The **language model for Wikipedia category** c_{w_i}*, smoothed with the entire Wikipedia as a background corpus* W*, is the probability distribution over words (or phrases) in the collection of all Wikipedia articles* $A_{c_{w_i}}$ *under the category* c_{w_i} *and its subcategories:*

$$P_{c_{w_i}}(w) = (1 - \lambda)P(w|A_{c_{w_i}}) + \lambda P(w|W) \tag{5.6}$$

Definition 5.6 ***Ranking of Wikipedia Category***
The **ranking of a Wikipedia category** c_{w_i} *for a given debate* D *is in descending order of:*

$$P(D|c_{w_i}) = \prod_{w_j \in D} P_{c_{w_i}}(w_j) \tag{5.7}$$

Note that these categories not only have the canonical facet name as a child, but may themselves be in a parent-child (or ancestor-descendant) relationship with each other. We retain these relationships as well, and later use them for the DAG structure of our **final topic graph***.*

5.4.2 Graph Coarsening Algorithm

At this point we could directly use the hierarchy of the selected Wikipedia categories (perhaps, with heuristics to remove occasionally occurring cycles) in order to impose a topic hierarchy over the identified Wikipedia articles. However, this would yield a fairly noisy topic structure, as Wikipedia often exhibits unsystematic diversity (by its grassroots contributors) and does not enforce terminological standards (not to speak of ontological structures). For example, topics like "radioactive waste", and "nuclear safety" are part of a larger debate on "renewable energies", but Wikipedia does not organize them in this manner. Therefore, and inorder to build our **final topic graph***, we devises a clustering method on the Wikipedia categories, with preservation of whatever parent-child relationships are already present among categories.*

To this end, we adopted and extended a graph coarsening *algorithm, originally developed for the different task of multi-level graph partitioning [63].*

Given a graph, a coarser graph can be obtained by collapsing adjacent vertices [63]. The edge between two vertices is collapsed and a multinode consisting of these two vertices is created. This edge collapsing idea is defined in terms of matchings.

Definition 5.7 ***Graph Matching***
A **matching of a graph** $G_i = (V_i, E_i)$*, is a set of edges, no two of which are incident on the same vertex. Thus, the next level coarser graph* G_{i+1} *is constructed from* G_i *by*

finding a matching of G_i *and collapsing the vertices being matched into multinodes. The unmatched vertices are simply copied over to* G_{i+1}.

A maximal matchings *are used to obtain the successively coarse graphs.*

Definition 5.8 ***Maximal Matching***
A matching is maximal if any edge in the graph that is not in the matching has at least one of its endpoints matched.

The number of edges belonging to the maximal matching may be different, depending on how matching are computed. The maximal matching *that has the maximum number of edges is called* maximum matching.

Definition 5.9 ***Maximum Matching***
A **maximum matching** *is a matching that contains the largest possible number of edges. There may be many maximum matchings. The matching number* $\nu(G)$ *of a graph* G *is the size of a maximum matching. Every maximum matching is maximal, but not every maximal matching is a maximum matching.*

There are many methods that can be used to select maximal matchings for coarsening. We describe briefly two of these methods.

- *Random Matching: The vertices are visited in random order. If a vertex* u *has not been matched yet, then one of its unmatched adjacent vertices is randomly selected. If such a vertex* v *exists, the edge* (u, v) *is included in the matching and mark vertices* u *and* v *as being matched. If there is no unmatched adjacent vertex* v*, then vertex* u *remains unmatched in the random matching. The complexity of the above algorithm is* $O(|E|)$.

- *Heavy Edge Matching: Finding a maximal matching that contains edges with large weight is the idea behind the heavy-edge matching. A heavy-edge matching is computed using a randomized algorithm similar to that for computing a random matching described earlier. The vertices are again visited in random order. However, instead of randomly matching a vertex* u *with one of its adjacent unmatched vertices,* u *is matched with the vertex* v *such that the weight of the edge* (u, v) *is maximum over all valid incident edges. The complexity of computing a heavy-edge matching is* $O(|E|)$.

5.4.2.1 Constructing Initial Topic Graph

Definition 5.10 ***Initial Topic Graph***
We construct a node- and edge-weighted directed **initial topic graph** $G_I = (V, E)$ *as follows. Each* $v_i \in V$ *corresponds to a Wikipedia category produced by the previously described method. Let the category corresponding to* v_i *be denoted by* $c(v_i)$*.* $E = \{(v_i, v_j)|c(v_i)$ *is a parent of* $c(v_j)\}$*.*
The node weight *of* v_i*, denoted by* $w(v_i)$ *is the number of distinct facets that were mapped to* $c(v_i)$ *or one of its descendants.*
The edge weight for an edge $e_{ij} = (v_i, v_j)$*, denoted* $w(e_{ij})$ *is proportional to the number of common Wikipedia articles under* $c(v_i)$ *and* $c(v_j)$ *(and their subcategories). Let* $A_{c(i)}$ *and* $A_{c(j)}$ *denote the Wikipedia articles under* $c(v_i)$ *(and its subcategories) and* $c(v_j)$ *(and its subcategories), then* $w(e_{ij}) = \frac{|A_{c(i)} \cap A_{c(j)}|}{|A_{c(i)} \cup A_{c(j)}|}$*, the Jaccard coefficient.*

So a category is considered more important if it transitively contains a large number of facets. We prune out very generic categories and also very sparse categories of exotic specificity, by using upper and lower bounds α *and* β *as thresholds for the category node weights.*

5.4.2.2 Constructing Final Topic Graph

From the initial topic graph constructed as described previously, we induce the final topic graph defined as follows.

Definition 5.11 ***Final Topic Graph***
A node- and edge-weighted DAG **final topic graph** $G_F = (V, E)$ *is a hypergraph constructed such that each* $v_i \in V$ *corresponds to a topic denoted by* $t(v_i)$*.* $E = \{(v_i, v_j)|t(v_i)$ *is a parent of* $t(v_j)\}$*. Each topic* $t(v_i)$ *represents a graph* $G_S = (V', E')$ *such that* G_S *is a subgraph of the initial topic graph* G_I *(*$G_S \subset G_I$*) and* G_S *is induced by* V' *(*$G_S = G_I[V']$*).* V' *corresponds to a set of highly correlated Wikipedia categories.*

To construct the final topic graph, we run the graph coarsening *algorithm of [63] on the initial topic graph, by gradually collapsing nodes and incident edges. For visiting the various edges of the graph during the process of identifying the clusters of nodes to collapse together, we consider the* globally-random locally-greedy *strategy (see Figure 5.3 for an example). This strategy visits the nodes of the graph in a random order and for each node* v*, it locally order the edges* $I(v)$ *that are incident on* v*. The graph*

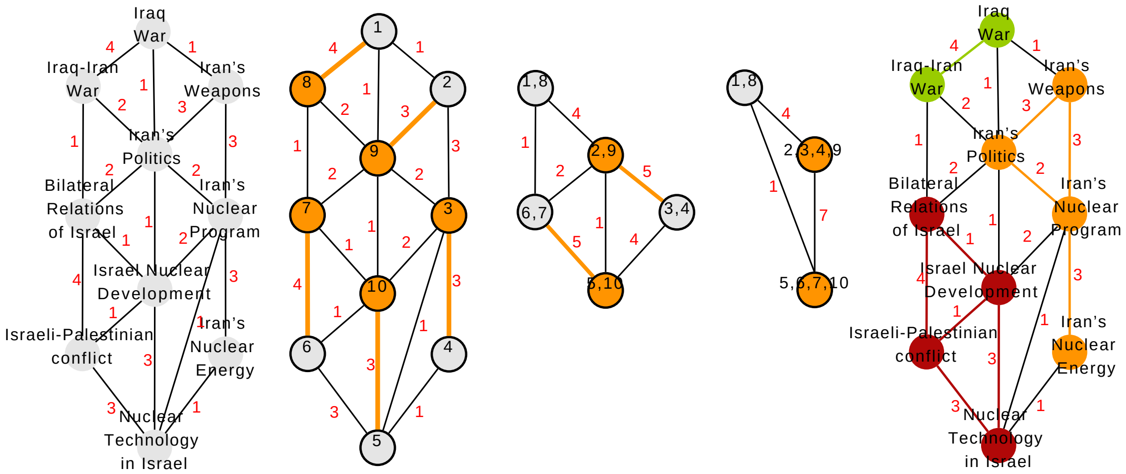


Figure 5.3: Globally-random locally-greedy strategy with heavy-edge matching

is then matched and coarsened with the heavy-edge *heuristic which prefers edges with higher weights. In our experiments, we ran the algorithm until it arrived at around 500 clusters, and we generated a label (topic) for each cluster based on the most frequent* n*-gram in its underlying set of facet strings.*

5.4.3 DAG Construction

The graph coarsening algorithm works only with undirected graphs, it ignores edge orientation. So now we need to impose a DAG structure among the final clusters. This is done in two steps.

5.4.3.1 DAG Construction Phase 1

First, we aggregate the parent-child relations for two clusters A *and* B *from their included categories* $a \in A$, $b \in B$. *For* a *being a parent of* b *in the original Wikipedia hierarchy, we compute the* ***average*** *of their edge weights into an aggregated weight* $w(A \sqsupset B)$. *For* a *being a child of* b *we analogously compute the aggregated* $w(A \sqsubset B)$. *Now the idea is to construct a parent-child edge from* A *to* B *if* $w(A \sqsupset B) > 0$ *and* $w(A \sqsubset B) = 0$, *and analogously for child-parent edges. Unfortunately, we cannot guarantee such conditions; we may have clusters with contradictory edges among their included categories.*

5.4.3.2 DAG Construction Phase 2

As a second step, we, therefore, construct a priority order among all cluster pairs and then proceed in a greedy manner. We sort cluster pairs (A, B) *in descending order of* $\mathtt{priority}(A, B) = w(A \sqsupset B) - w(A \sqsubset B)$. *Now we construct parent-child edges between clusters in this order as long as* $\mathtt{priority}(A, B) > 0$. *While doing this, we check for possible formation of cycles, and drop edges accordingly. Subsequently, we repeat this procedure for the dual priority order* $w(A \sqsubset B) - w(A \sqsupset B)$ *if it leads to additional edges without creating cycles. The greedy heuristic in this approach helps to ensure that the most prominent relationships between clusters are captured.*

The outlined algorithm is our way of constructing a topic DAG from Wikipedia categories, and completely mapping all collected facets onto these topics.

Alternative approaches are conceivable, for example, employing standard clustering methods at the level of facets, debates, Debatepedia topics, Wikipedia articles, or categories. Among such alternatives, the one that looks most intriguing would be to run a hierarchical agglomerative clustering (HAC) algorithm on the category candidates that we already have before the graph coarsening step. We compare our technique against such an approach in the experimental evaluation, and will point out limitations and drawbacks of the HAC approach.

Figure 5.2 shows an example of what the hierarchy looks like in the context of the recent Arab Spring.

5.5 Experimental Evaluation

We are interested in comparing our elaborate method for mapping facets to topics and for constructing the topic DAG against alternatives like hierarchal clustering.

5.5.1 Experimental Setup

The topic DAG considered for this evaluation study consists of 23,000 facets, 4,000 canonicalized facets mapped to 5,000 Wikipedia categories, and 500 controversial topics organized in a DAG with ca. 3,000 topic-subtopic edges. One of our judges evaluated a randomly selected set of 500 facets. The 500 facets have 413 canonicalized facets, 387 Wikipedia categories, and 166 topics in the topic DAG. More precisely,

1. *For each facet we present its canonical facets computed by the method described in Section 5.3 (**LM-DebRank**),*

2. *For each of the corresponding canonical facets, we presents the corresponding Wikipedia categories computed by the method described in Section 5.4.1 (**LM-CatRank**),*

3. *The corresponding controversial topic of each category computed by the coarsening algorithm (**CO**) of Section 5.4.2).*

Judges were asked to choose relevant:

1. *Canonical facets for raw facets,*

2. *Wikipedia categories for canonical facets,*

3. *Topics for Wikipedia categories.*

*In addition, the judges were asked to assess the correctness of a random set of 276 topics parent-child pairs from the topics in the final topic DAG. These topic pairs were collected using our DAG construction algorithm (**DAG**) described in Subsection 5.4.3. The main measure of interest is the precision of the assessed mappings.*

5.5.2 Results

*Table 5.3 shows the results. We found that about **69%** of the raw facets* f *were assigned correctly to at least one of the canonicalized facets* F*, and **78%** of the presented* F *were correctly assigned to at least one of the Wikipedia categories* C*. Wrong* (f, F) *pairs were found mainly for very general facets (e.g., "the bill"). In such cases, informative LMs for facets require longer contexts. For the* (C,T) *pairs obtained by the **(CO)** method, **73%** were correct, and **66%** of the facets have at least one correct complete path (*$f \rightarrow F \rightarrow C \rightarrow T$*). For the* (T, T) *pairs in the final topic DAG, **69%** were found to be correct, using the (**DAG**) approach.*

*As an alternative approach to both the (**CO**) and (**DAG**), we ran a hierarchical agglomerative clustering **(HAC)** algorithm of the Weka package[1] on the collected Wikipedia categories. Hierarchical clustering produces hierarchical representations in which the clusters at each level of the hierarchy are created by merging clusters at the next lower*

[1] www.cs.waikato.ac.nz/ml/weka

Items (#)	Pairs (#)	Approach	Accuracy
f **(500)**	(f,F) **(1500)**	LM-DebRank	0.69
F **(413)**	(F,C) **(1239)**	LM-CatRank	0.78
C **(387)**	(C,T) **(387)**	CO	0.73
nodes (137)	(T,T) **(276)**	DAG	0.69
–	**–**	CO&DAG	0.50
C **(387)**	(C,T) **(387)**	HAC-CT	0.68
nodes (532)	(T,T) **(276)**	HAC-TT	0.71
–	**–**	HAC-CT&HAC-TT	0.48

Table 5.3: Results for controversial topics graph

level. At the lowest level, each cluster contains a single observation. At the highest level there is only one cluster containing all of the data. There are two strategies for hierarchical clustering: agglomerative (bottom-up) and divisive (top-down). Agglomerative strategies start at the bottom and at each level recursively merge a selected pair of clusters into a single cluster. This produces a grouping at the next higher level with one less cluster. The pair chosen for merging consist of the two groups with the smallest intergroup dissimilarity. Divisive methods start at the top and at each level recursively split one of the existing clusters at that level into two new clusters. The split is chosen to produce two new groups with the largest between-group dissimilarity. With both strategies there are $N-1$ *levels in the hierarchy [50].*

With hierarchical clustering, about ***68%*** *of the 387 random* (C,T) *pairs were correct (**HAC-CT** vs. **CO**), while the 276 random* (T,T) *pairs have* ***71%*** *accuracy (**HAC-TT** vs. **DAG**). Since the construction of topics hierarchy depends on the construction of each topic cluster, the precision of these two components togother is computed. The precision of **CO** and **DAG** is higher than the precision of **HAC-CT** and **HAC-TT**. Also when comparing the judiciously constructed DAG vs. standard HAC, one also needs to consider the overall size and structure of the resulting graphs. DAG produces a compact and nicely explorable hierarchy with a total of 533 nodes and 299 edges (so that most nodes are leaves in the DAG). In contrast, HAC creates a much larger binary tree with 1056 nodes and 1055 edges in total. This graph is much more tedious to explore, and does not convey the same level of informativeness as our constructed DAG.*

5.6 Related Work

A method for automatically deriving a hierarchical organization of concepts from a set of documents is presented in [107]. Salient words and phrases extracted from the documents are organized hierarchically using a type of co-occurrence known as subsumption.

[30] presents an approach to the automatic acquisition of taxonomies or concept hierarchies from a text corpus. The approach follows Harris' distributional hypothesis and model the context of a certain term as a vector representing syntactic dependencies which are automatically acquired from the text corpus with a linguistic parser. On the basis of this context information, a lattice is produced and converted into a special kind of partial order constituting a concept hierarchy.

An approach is described in [27] that organizes extracted features in a hierarchical fashion. It includes product information in a user-defined taxonomy of features to streamline and better organize the learned features. The basic idea is to employ similarity matching techniques together with WordNet to map learned features into this user-provided taxonomy.

[121] extends both PLSA and LDA to induce multi-grain topics from review corpus. The models generate terms from either a global topic, which is chosen based on the document level context, or a local topic, which is chosen based on a sliding window context over the text. The local topics model features that are rated throughout the review corpus.

[49] performs multilevel latent semantic analysis to group features expressions. At the first level, all the words in features expressions are grouped into a set of concepts using LDA. The results are used to build latent topic structures for features expressions, e.g., touch screen: topic-1, topic-2. At the second level, features expressions are grouped by LDA again according to their latent topic structures produced from the first level and context snippets in reviews.

[125] uses a full tree ontology to denote the relationships of features of a product. The leaves of the tree are positive or negative sentiments. It then uses a hierarchical classification model to learn to assign a sentiment to each node, which is reflected as a child leaf node. Hierarchical classifier considers parents when classifying children. However, the ontology for each product has to be built manually.

A variety of similarities are used in [137] to cluster features expressions into features categories. Two types of constraints are extracted automatically and incorporated into the topic modeling method LDA to produce a semi-supervised LDA method. These constraints are applied to group product features.

5.7 Summary

In this chapter, we presented an approach for imposing a topic hierarchy on the fine-grained topics derived from Debatepedia. We used Debatepedia as a bootstrapping asset to focus on controversies, but eventually map facets and quotations to Wikipedia categories for a richer organization of topics. As an alternative approach , we used a hierarchical agglomerative clustering algorithm on the collected Wikipedia categories. The precision of the mapping approach was experimentally found to be higher than the precision of the hierarchal approach.

Chapter 6

Political Support/Oppose Lexicon

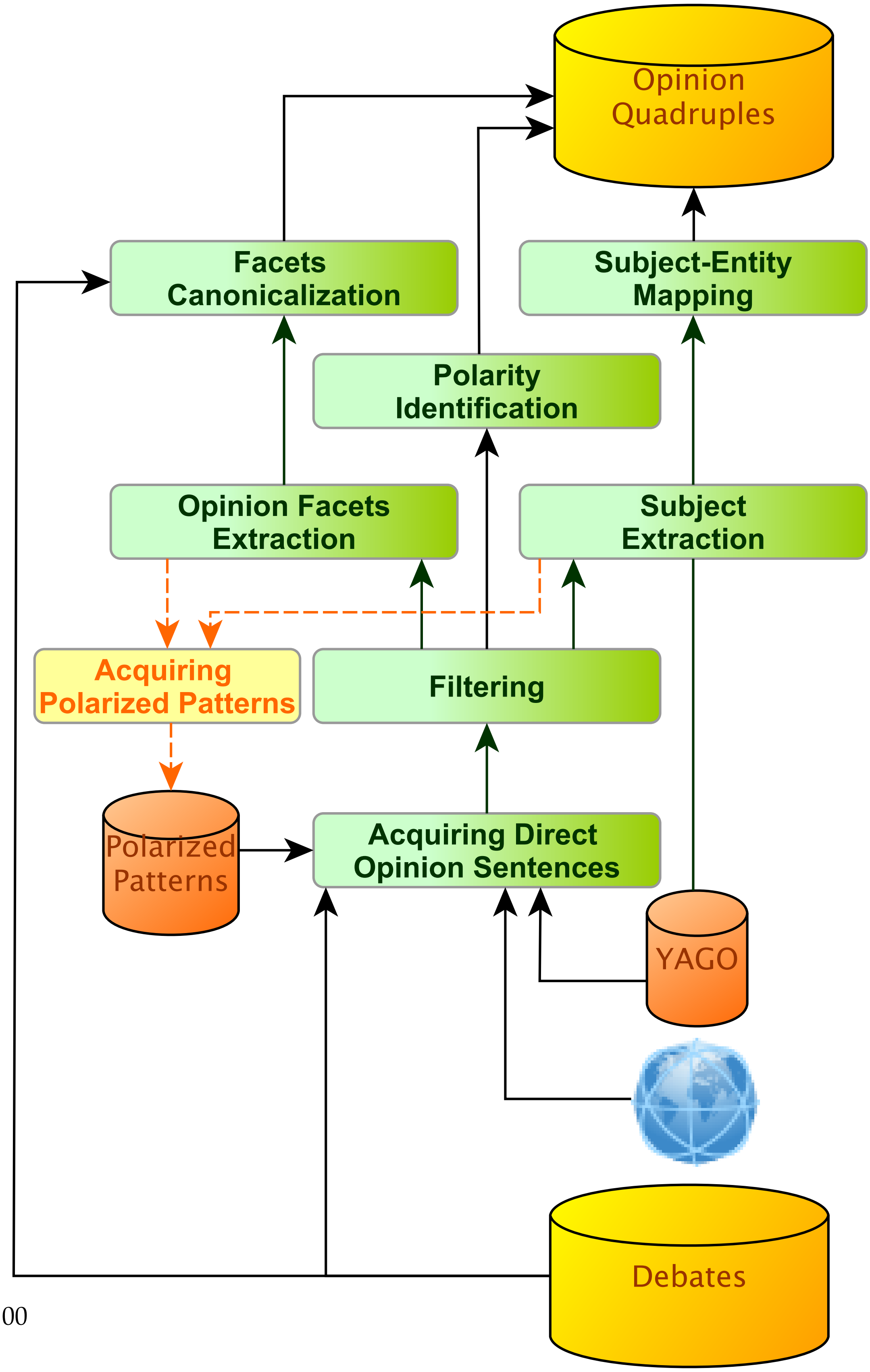
6.1 Introduction

As it was described in Chapter 3, we used a set of polarized patterns in order to extract opinions from news articles. However, the number of opinions are limited by the initial polarized patterns set, since only those patterns are queried. For example, we find "supports", but we would miss "is in favor of". Clearly, support and opposition to something can be expressed in several different and would occur in different textual variations (for example, "strongly disapproved" would indicate opposition). Therefore, our next task is to automatically *identify more support and oppose patterns and create a lexicon of phrases. These phrases can then be used to collect more snippets from the Web and extract opinions from them (see Figure 6.1 for an overview).*

In this Chapter we present an unsupervised method in Section 6.2, supervised methods in Section 6.3, and a semi-supervised method in Section 6.4, developed to automatically create a larger lexicon of political opinions expressions. Section 6.5 presents our experimental evaluation. Section 6.6 discusses related work. We conclude with Section 6.7.

6.2 Unsupervised Approach

Our first approach is an unsupervised method, bootstrapped from the initial polarized patterns set.


Opinion
Quadruples
Facets
Canonicalization
Subject-Entity
Mapping
Polarity
Identification
Opinion Facets
Extraction
Subject
Extraction
Acquiring
Polarized Patterns
Filtering
Polarized
Patterns
Acquiring Direct
Opinion Sentences
YAGO
Debates

6.2.1 Candidate Sentences

*We identify 10 support triples and 10 oppose triples from our initial set of triples extracted using the initial polarized patterns. From each triple, we isolate the opinion holder and the opinion target. Here we use the raw facets and entity names (not the canonicalized ones) because these are close to the variety of expressions used in news sources. Each ⟨entity name, facet⟩ pair is now viewed as a query submitted to a search engine. Examples of the resulting snippets are shown in Table 6.1. The underlined text indicates phrases which could be used to populate the lexicon of support phrases. A dependency parser is used to parse snippets. The parsing step is required to filter out all snippets that do not follow the ⟨**SPO**⟩ structure as it was described in Section 3.2.1 of Chapter 3. Table 6.2 gives an example of a candidate sentence that follows this structure.*

Obama **strongly defends** US Military Action in Libya
Barack Obama **defended his decision to** launch military action in Libya
President Obama said Saturday that he **had authorized** limited military action in Libya
President Obama tells Americans that he **ordered** military action in Libya

Table 6.1: Examples for snippets extracted for ("Obama", "military action in Libya") pairs

6.2.2 Candidate Verb Phrases

*The **verb phrase** between the entity name and the facet in a candidate snippet is isolated (e.g., "strongly defends", "defended his decision to", etc.). We lookup the verb that appears in both the subject relation and the object relation of the typed dependency of a snippet. The identified verb can be part of a verb phrase. For example the sentence in Table 6.2, has the typed dependency relations "**nsubj(defends-3, Obama-1)**" and "**nsubj(defends-3, Action)**". This indicates that the verb we are interested in is "defends" which is part of a verb phrase. Therefore using the parsed tree of the sentence, we target the top parent verb phrase that includes the verb we are interested in but excluding from it the noun of phrase which represent the object. In the parse tree, a verb phrase is denoted as "**VP**". For example, the parse tree of the sentence in Table 6.2, indicates that the top parent verb phrase of the verb "defends" is "strongly defends US Military Action in Libya". We exclude the noun phrase of the object "US Military Action in Libya", and this gives us the phrase "strongly defends".*

Sentence:

Obama strongly defends ***the US military action in Libya***

Parse Tree:

```
(ROOT
  (S
    (NP (NNP Obama))
    (VP
      (ADVP (RB strongly))
      (VBZ defends)
      (NP
        (NP (NNP US) (NNP military) (NNP action))
        (PP (IN in)
          (NP (NNP Libya)))))))
```

Typed Dependencies:

nsubj(defends-3, **Obama**-1)

advmod(defends-3, strongly-2)

root(ROOT-0, defends-3)

nn(action-6, US-4)

nn(action-6, military-5)

dobj(defends-3, ***action***-6)

prep(action-6, in-7)

pobj(in-7, Libya-8)

Table 6.2: Candidate snippets for bootstrapping the initial polarized patterns

6.2.3 Candidate n-grams

The candidate verb phrases can still contain too much noise. Therefore, we generate all n*-grams (for* $n = 1$ *to* 5*) for the collected substrings, and each* n*-gram becomes a candidate. The* n*-grams are ranked by their occurrence frequency in the collected snippets. We retain the top-*k n*-grams.*

Initially, we made the assumption that, if a query is formed by a "support" triple, then all candidate n*-grams are candidates for the support set. However, the same ⟨entity name,facet⟩ query formulated from a support triple could also return an "oppose"* n*-gram. Therefore, we exclude all* n*-grams that appear in both the "support" and the "oppose" candidates sets, if the differences between their occurrence frequencies in the two sets are less than the given threshold, while retained* n*-grams are added to the sets they appear in with higher occurrence frequencies.*

Examples of some of the collected n*-grams are shown in Table 3.10.*

6.3 Supervised Approaches

We train two supervised classifiers. A linear **SVM** *classifier which is implemented using the SVMlight software package*[1]*), and a J48 Decision Tree* **(DT)** *classifier of the Weka package*[2]*. We consider as basic features the* n*-grams of the* verb-phrases ***(NGRAMS)*** *as it was described in Section 6.2. We also explore additional features and their combinations as inputs to the different supervised approaches.*

6.3.1 Discriminative Strength

This feature considers the candidate n*-grams of the verb-phrases and the ⟨entity name, facet⟩ pairs of the* support *triples and the* oppose *triples used to collect candidate verb-phrases. The number of times an* n*-gram* n *co-occurs with a* support pair s*, and the number of times it co-occurs with an* oppose pair o *are indicators of whether* n *is a support or an oppose phrase. We formalize this approach into a measure as follows.*

Definition 6.1 ***Discriminative Strength***
Let $S = \{s_1, s_2, \cdots, s_l\}$ *be the set of known support pairs and let* $O = \{o_1, o_2, \cdots, o_m\}$

[1] svmlight.joachims.org
[2] www.cs.waikato.ac.nz/ml/weka

be the set of known oppose pairs. Then, the ***Discriminative Strength (*** DS***)*** *of an* n*-gram* n *is:*

$$\mathsf{DS}(n) = \frac{\mathsf{MI}(S, n) - \mathsf{MI}(O, n)}{\mathsf{MI}(S, n) + \mathsf{MI}(O, n)} \tag{6.1}$$

where $\mathsf{MI}(S, n)$ *is the* mutual information *of* S *and* n.

Definition 6.2 ***Mutual Information***
The mutual information of support pairs S *and an* n*-gram* n *is computed as follows.*

$$\mathsf{MI}(S, n) = \sum_{i=1}^{l} P(s_i, n) \log \frac{P(s_i, n)}{P(s_i)P(n)} \tag{6.2}$$

where $P(s_i, n)$ *is the probability that* s_i *and* n *co-occur.*

Definition 6.3 ***Pair and*** n***-gram Likelihood***
The probability that a support pair s_i *and an* n*-gram* n *co-occur is estimated as follows.*

$$P(s_i, n) = \frac{\mathsf{hits}(s_i, n)}{\mathsf{hits}(s_i)\mathsf{hits}(n)} \tag{6.3}$$

where $\mathsf{hits}(s_i, n)$ *is the number of hits returned by the search engine for a query formulated as "*s_i *AND* n*".*

If $\mathsf{DS}(n)$ *is positive, it is more likely that* n *is a support phrase, otherwise, it is an oppose phrase.*

6.3.2 Synonyms and Antonyms

We add synonyms ***(SYN)*** *to each* n*-gram in the features set. Similarly, by substituting antonyms* ***(ANT)*** *in the verb-phrase* n*-grams, we enhance the classifier in its ability to learn the opposite opinion. For example, for the* n*-gram "takes steps to change", we identify verbs "takes" and "to change". Adding synonyms to "to change", we arrive at a features set that includes "takes steps to alter", "takes steps to modify", etc. Adding antonyms of "to change", gives us "takes steps to continue", "takes steps to keep", etc..*

6.4 Semi-Supervised LM-based Classifier

We developed a nearest-neighbor-style classifier based on statistical language models (LMs). In this approach, we construct LMs for both the support and the oppose verb-phrases as follows.

Definition 6.4 ***The Support Language Model***
Let n^+ denotes the set of all **support** n**-grams** *computed for all known* support *verb-phrases. The* **LM of** n^+*, using the set of n-grams in n^+ and the set of n-grams in a background corpus N used for smoothing, is defined as the following probability distribution over n-grams:*

$$P_{n^+}(w) = (1-\lambda)P(w|n^+) + \lambda P(w|N) \tag{6.4}$$

where w is an n-gram of n^+, $P_{n^+}(w)$ is the estimated probability of the w in the LM of n^+, $P(w|n^+)$ is the probability of w in n^+ and $P(w|N)$ is the probability of w in N.

Definition 6.5 ***The Oppose Language Model***
Let n^- be the set of all **oppose** n**-grams** *computed for all known* oppose *verb-phrases. The* **LM for** n^-*, using n^- itself and the set of n-grams in N for smoothing, is defined as:*

$$P_{n^-}(w) = (1-\lambda)P(w|n^-) + \lambda P(w|N) \tag{6.5}$$

Let Q be a newly seen n-gram. We now construct two queries as follows. A synonym query $Q_{syn} = \{SYN_Q\}$ *and an* antonym query $Q_{ant} = \{ANT_Q\}$*, where SYN_Q and ANT_Q are constructed using the synonyms and the antonyms n-grams as described in Section 6.3.2.*

Definition 6.6 ***Synonym Query Likelihood***
The probability of generating the synonym query Q_{syn} *given* the support LM *is:*

$$P(Q_{syn}|n^+) = \prod_{w_j \in Q_{syn}} P_{n^+}(w_j) \tag{6.6}$$

Analogously, the probability of generating the synonym query Q_{syn} *given* the oppose LM *is:*

$$P(Q_{syn}|n^-) = \prod_{w_j \in Q_{syn}} P_{n^-}(w_j) \tag{6.7}$$

Both $P_{n^+}(w_j)$, and $P_{n^-}(w_j)$ are estimated based on the value of an aggregation function (e.g., average sum (SUM), or maximum value (MAX)) of the Jaccard similarity between w_j and any n-gram in n^+ or n^-, respectively. Similarly, we compute the two corresponding likelihoods of the antonym query Q_{ant}.

The newly seen n-gram is then classified based on the highest value of two measures: the support weight ***(SW)*** *and* the oppose weight ***(OW)****. The intuition behind these two weights is that if the probability of generating the synonyms of an n-gram given* *the support/oppose LM is high, then most probably the probability of generating the antonyms of the* n*-gram given the oppose/support LM is also high and vice versa.*

$$SW(Q) = P(Q_{syn}|n^{+}) + P(Q_{ant}|n^{-}) \tag{6.8}$$

$$OW(Q) = P(Q_{syn}|n^{-}) + P(Q_{ant}|n^{+}) \tag{6.9}$$

6.5 Experimental Evaluation

Overall, we collected approximately 8,000 snippets from the Web using 10 support triples and 10 oppose triples. About 2,000 n*-grams were computed from the verb-phrases extracted from these snippets (1119* n*-grams from the support snippets and 936* n*-grams from the oppose snippets).*

For the unsupervised approach described in Section 6.2, our judges evaluated 460 n*-grams out of the collected* n*-grams (top 230* n*-grams from the support* n*-grams, and top 230* n*-grams from the oppose* n*-grams).*

For the (semi-)supervised approaches, 540 n*-grams were manually labeled. Out of the 540* n*-grams, 400* n*-grams used for training (200 support* n*-grams and 200 oppose* n*-grams), and 140* n*-grams (70 support* n*-grams and 70 oppose* n*-grams) used for testing the different approaches (**SVM, DT, LM**) using different combinations of features (**NGRAMS, DS, SYN, ANT**).*

The main measures for the unsupervised approach are relative precision *and* relative recall *(based on the 460 samples). Let* TP *denotes the number of* n*-grams that were* correctly *classified as* support, *while* FP *is the number of* n*-grams that were* incorrectly *classified as* support. *Let* FN *be the number of* n*-grams that were* incorrectly *classified as* oppose, *but should have been classified as* support. *The* precision *and* Relative recall *are then computed as:*

$$Precision = \frac{TP}{TP + FP} \tag{6.10}$$

$$Relative\ Recall = \frac{TP}{TP + FN} \tag{6.11}$$

Precision *and* recall *are used for the (semi-)supervised approaches (based on the withheld 140 test cases).*

*Table 6.3 shows the best results for the different methods using different combinations of features. The unsupervised approach outperforms the other methods, despite its simplicity. The LM method performs better than DT and SVM. Using n-grams only as features gives bad results because of data sparseness. For this reason, we used aggregated features (e.g., **SUM**, **MAX**) over synonyms and antonyms features. Using synonyms only in the similarity measures gives better results than using antonyms as well while results with the SUM as an aggregation function outperform the results of the MAX function.*

Approach	**Precision**	**Recall**
Unsupervised	**0.79**	**0.60**
SVM+NGRAMS+DS	0.47	0.21
SVM+SYN+SUM	0.61	0.60
DT+SYN+SUM	**0.63**	**0.62**
LM+SYN+SUM	**0.69**	**0.70**

Table 6.3: Results for support/oppose classifiers

6.6 Related Work

To generate opinion lexicons, there are two automated approaches: a dictionary-based approach and a corpus-based approach.

6.6.1 Dictionary-Based Opinion Lexicons

This approach is based on bootstrapping using a small set of seed opinion words and an online dictionary (e.g., WordNet). A small set of words with known polarities are grown using WordNet synonyms and antonyms. The newly found words are added to the seed list. The next iteration starts. The iterative process stops when no more new words are found [36, 46]. Additional information (e.g., glosses) in WordNet and additional techniques (e.g., machine learning) can also be used to generate better lists [2, 41, 42, 43]. The dictionary based approach is unable to find opinion words with domain specific polarities.

6.6.2 Corpus-Based Opinion Lexicons

The methods in the corpus-based approach uses syntactic or co-occurrence patterns and also a seed list of opinion words to find other opinion words in a large corpus [70, 48, 13, 98, 54, 102, 103, 123, 126, 6]. The key idea is to use a list of seed opinion adjective words, and a set of linguistic constraints or conventions on connectives to identify additional adjective opinion words and their polarities [51]. Constraints are designed for connectives, "and", "or", "but", "either or", and "neither nor". Learning using the log-linear model is applied to a large corpus to determine if two conjoined adjectives are of the same or different polarities. Same and different polarities links between adjectives forms a graph. Clustering is then performed on the graph to produce two sets of words: positive and negative.

In [102] a method called double propagation is proposed to extract domain specific sentiment words from reviews using some seed opinion words. It exploits certain syntactic relations of opinion words and object features for extraction. It has been found that opinion words can be recognized by identified features, and features can be identified by known opinion words. The extracted opinion words and features are utilized to identify new opinion words and new features, which are used again to extract more opinion words and features.

[36] explores the idea of intra-sentential and inter-sentential sentiment consistency. It proposes to consider both opinion words and object features together, and use the pair ⟨object feature, opinion word⟩ as the opinion context. The method determines opinion words and their polarities together with the object features that they modify. The methods described in [117, 122] find context specific opinions based on syntactic POS patterns rather than object features and opinion words that modify them. The Conditional Random Fields method [74] is used in [24] for extracting opinion expressions, which can have any number of words.

6.7 Summary

In this Chapter we presented unsupervised, semi-supervised, and supervised methods, developed to automatically create a larger lexicon of political opinions expressions. Our experimental evaluation shows that the unsupervised approach, despite its simplicity, is effective.

Chapter 7

OpinioNetIt System

7.1 Introduction

Our system, called "OpinioNetIt", builds an opinion-base, which is both structured*, as well as* faceted*. Opinions in our system are represented as RDF triples of the form ⟨opinion holder, opinion, opinion target⟩. Additionally, each triple is augmented with a fourth component, the* context *from which the triple was extracted. This structured representation allows users to query the system with SPARQL. Moreover our RDF opinion-base allows faceted browsing.*

OpinioNetIt makes use of two distinct sources to extract opinions: i) text snippets, carefully chosen from online news sources such as CNN, Aljazeera, BBC, etc., and, ii) quotations that are also extracted from online news articles about different political controversial topics. Our approach and methodology were described in the previous chapters. In this chapter, we focus on the architecture of OpinioNetIt and describe how our system can be used for different kinds of political opinions analysis. OpinioNetIt relies on two main components: "Acquiring Opinions", and "Organizing Topics" (see Figure 7.1 for an overview). The "Acquiring Opinions" component uses a combination of techniques to extract opinions from online news sources. The "Organizing Topics" component crawls and extracts debates from online sources such as Wikipedia and Debatepedia, and organizes these debates in a hierarchy of political controversial topics.

OpinioNetIt can be used for different kinds of applications which require political analysis. Here, we specifically focus on three kinds of applications: i) identifying flip-floppers (those who repeatedly change their minds on a topic), ii) heat map analysis indicative of political bias (news media outlets reporting stories with a certain bias), and, iii) dissenters (those who deviate from "expected" opinions).

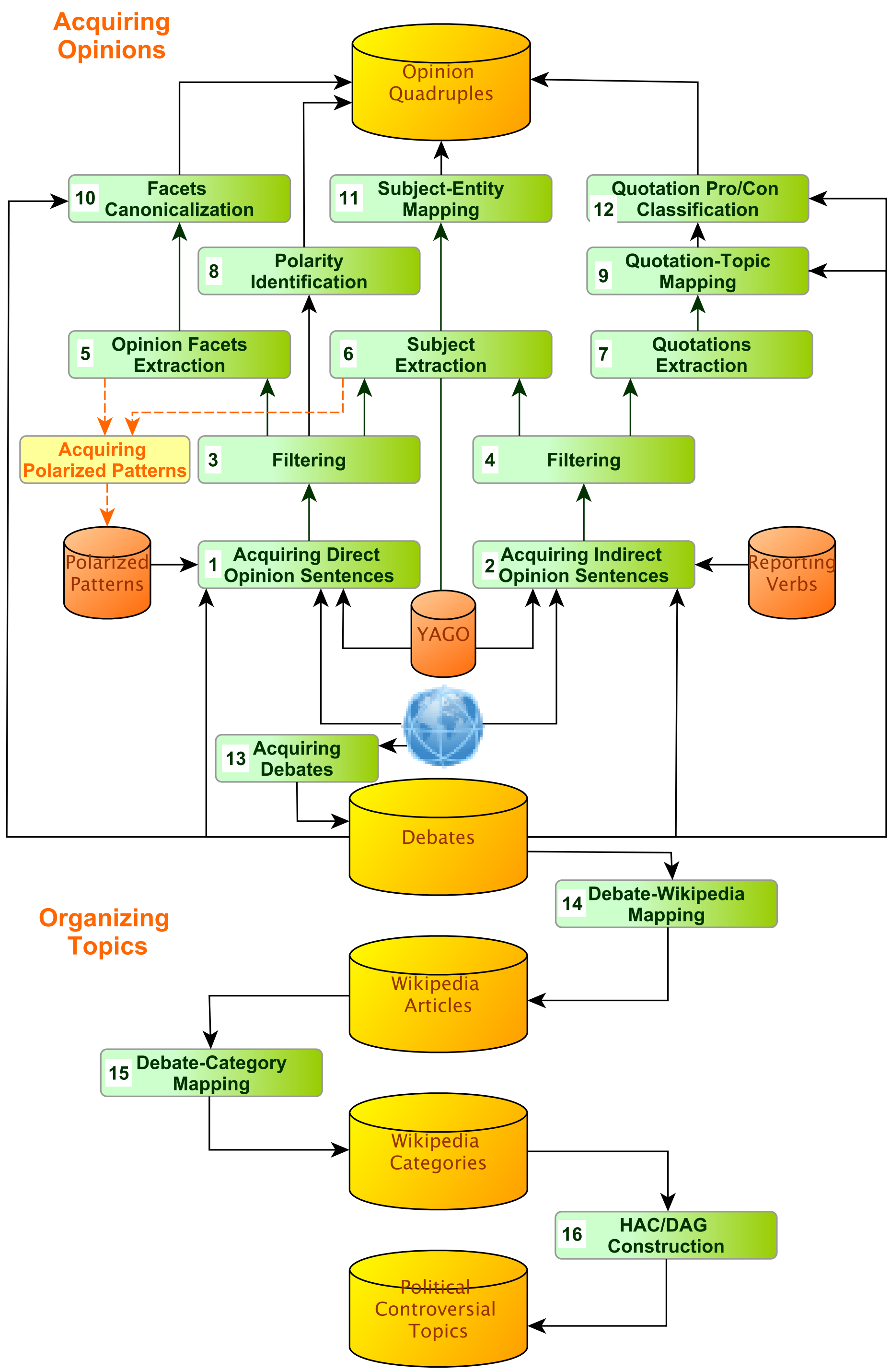


Figure 7.1: System architecture.

The rest of this chapter is organized as follows. Section 7.2 presents the system implementation. Section 7.4 introduces some use scenarios of the opinions network OpinioNetIt. Finally, in Section 7.5, we discuss some related works.

7.2 System Implementation

We have implemented OpinioNetIt in Java. Our system consists of two main components as shown in Figure 7.1.

Acquiring Opinions *This component consists of a number of subcomponents. Components (1) and (2) use web search engines such as Google and Bing to collect candidate opinion sentences. The extraction subcomponents (5), (6), and (7) use the dependency parser and NER tagger of the Stanford NLP package (SNLP) [71] in order to extract the opinions' targets and the opinions' holders from the set of snippets collected from online news resources. The mapping subcomponents (9) and (10) are responsible for mapping the extracted facets and quotations to fine-grained topics represented by the different debates. It uses a multinomial LM-based classifier over the n-grams features of the facets, the quotations and the debates. The pro/con classification subcomponent (12) uses both the Wordnet and the SNLP to expand and enrich the quotations' features by adding synonyms and antonyms. It implements several LM-based classifiers for the pro/con classification of quotations. The subcomponent (11) uses Yago to map each candidate opinion holder to a canonical name.*

Organizing Topics *Four subcomponents form the body of this component: The crawl subcomponent (13) crawls the available online resources (e.g. Wikipedia, Debatepedia) to form the set of debates to which opinion sentences should be mapped. It uses web search engines such as Google and Bing to collect the debates. The subcomponents (14), (15), and (16) construct the controversial topics hierarchy, making use of different online resources (e.g. Wikipedia categories, Debatepedia categories), and using different clustering algorithms (e.g. Metis [63]).*

All the outputs, opinions, facets clusters, and the opinions seeds are saved in a relational database. Opinions are saved in the RDF+text format, where the text consists of the contexts from which an opinion triple was extracted.

7.3 OpinioNetIt Web Interface

The Web-based frontend of OpinioNetIt runs AJAX for asynchronous communication with the server. Figure 7.2 is a screenshot of our system. The screenshot shows two main tabs on the left side: (1) the "Query Opinions" tab, and (2) the "Collect Opinions" tab, and on the right side (3) the "Opinions" board.

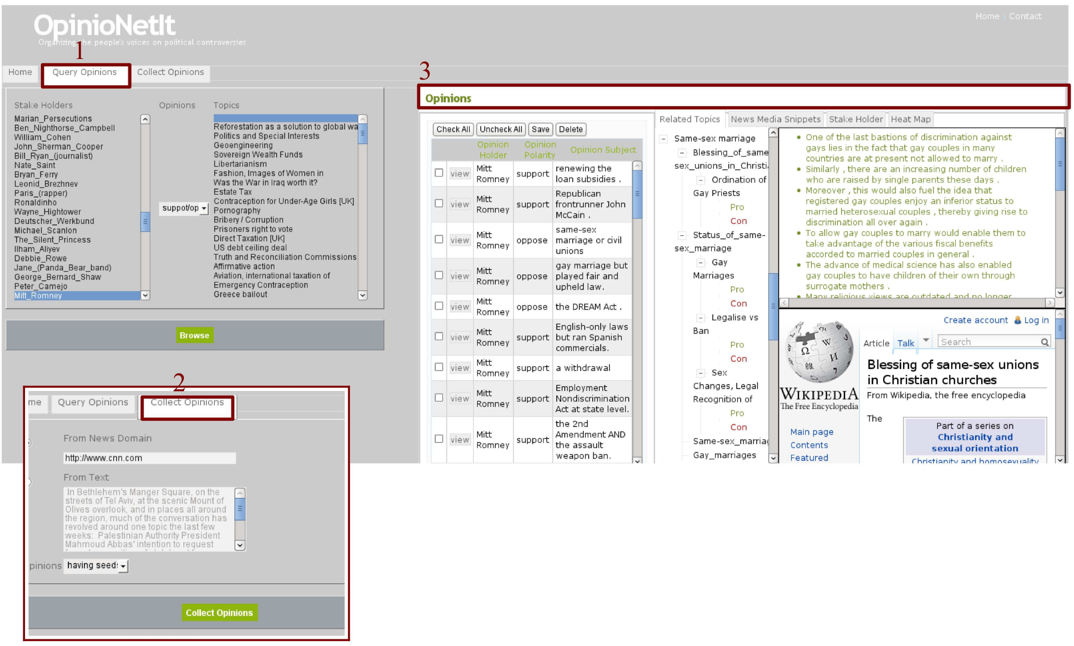


Figure 7.2: A screenshot of OpinioNetIt.

Query Opinions *tab (1) in Figure 7.2 allows user to query OpinioNetIt. User can search opinions by one or multiple persons, by one or multiple topics, by opinions or by any combination of the persons, topics, and opinions.*

Collect Opinions *tab (2) in Figure 7.2 allows users to interactively collect opinions from online news sources or from textual content. Users have the option to specify the seeds or the quotations or both in the extraction process.*

Opinions *tab (3) in Figure 7.2 shows the opinions resulted from the query issued through tab (1) or those resulted from the opinion collection process issued through tab*

(2). The opinion are listed in a table on the left side of the "Opinions" board. These opinions can further be browsed. A user clicks on the view link of each opinion to have more details presented in different tabs on the right side of the board. The following tabs can be browsed for each selected opinion:

(1) Related Topics: *A list of the Debatepedia pages, and the Wikipedia pages to which the opinion subject was mapped. The list allows the user to browse the hierarchy of the controversial topics. There are many scenarios in which this tab can be used. User clicks on a topic and view the pro and con arguments of that topic or finds relevant topics (e.g fine-grained topics, facets, quotations),*

(2) News Media Snippets: *A list of the supporting evidence for the selected opinion. These can be snippets or quotations extracted from news sources,*

(3) Stakeholder: *The Wikipedia page of the opinion holder.*

(4) Heatmap: *In case a user queries for the opinions of more than one opinion holder on more than one topic, a heatmaps is generated for the opinion holder's opinions on the different topics.*

For example in Figure 7.3, a user queries the opinions of "Barack Obama" through the "Query Opinions" tab. Barack Obama's opinions are listed in the table on the left of the "Opinions" board. Related topics, and news media snippets with respect to the clicked opinion in the table are shown in their corresponding tabs. Also Barack Obama's Wikipedia page is presented in the "Stake Holder" tab.

7.4 Use Cases

Beyond simple browsing, OpinioNetIt allows many interesting applications to built. As use cases for knowledge discovery and political analysis, we discuss some of these scenarios in the following:

7.4.1 Flip-flop Detection

User can discover opinion holders who flip-flop. A flip-flop, according to Wikipedia, is a sudden, real or apparent change of policy or opinion by a public official. To detect flip-flops, user groups opinions collected by our system by pairs of a stakeholder and a topic, If a group contains both support and oppose opinions, we may consider the politician to have flip-flopped (see Figure 7.4 for an example showing such opinions).

Figure 7.3: "Barack Obama" opinions along with the related tab views.

7.4.2 Dissenters Detection

OpinioNetIt can detect interesting deviations from correlated opinions on different topics. Often, once you know that a person supports topic A, it is clear that she or he also supports topic B, given the semantic connection between A and B, or simply the political affiliation or ideology of the person. To identify such topic pairs, we compute correlation coefficients and select the most positively correlated pairs. Our system can find notable dissenters: people who deviate from this pattern and support only one of A or B and oppose the other one. Table 7.1 shows some interesting examples that we found in our collection (dominant opinions: "++", "+-", "-+", or "–" with "+" denoting support and "-" denoting oppose).

7.4.3 Heatmaps

Users can browse the opinions of a set of stakeholders on different topics and compare their opinions. To ease this comparative study, we generate heatmaps of the opinion holder's opinions on the different topics. The stakeholders can be politicians, political

Topics Pair	Emissions Trading & Offshore Drilling
Dominant Patterns	(+−) (−+)
Examples	⟨Cynthia McKinney⟩ ⟨+⟩ ⟨Carbon Neutrality⟩ ⟨Cynthia McKinney⟩ ⟨-⟩ ⟨Offshore drilling⟩
Dissenter Patterns	(++) (−−)
Examples	⟨John McCain⟩ ⟨+⟩ ⟨Emissions trading⟩ ⟨John McCain⟩ ⟨+⟩ ⟨Offshore drilling⟩
Topics Pair	George W. Bush Politics & Abortion
Dominant Patterns	(+−) (−+)
Examples	⟨Sarah Palin⟩ ⟨+⟩ ⟨George W. Bush⟩ ⟨Sarah Palin⟩ ⟨-⟩ ⟨Abortion⟩
Dissenter Patterns	(++) (−−)
Examples	⟨Joe Biden⟩ ⟨+⟩ ⟨George W. Bush⟩ ⟨Joe Biden⟩ ⟨+⟩ ⟨Abortion rights⟩
Topics Pair	National Defense Authorization Act (NDAA) & Guantanamo Bay
Dominant Patterns	(++) (−−)
Examples	⟨Mitt Romney⟩ ⟨+⟩ ⟨NDAA⟩ ⟨Mitt Romney⟩ ⟨+⟩ ⟨Guantanamo Bay⟩
Dissenter Patterns	(+−) (−+)
Examples	⟨Barack Obama⟩ ⟨+⟩ ⟨NDAA⟩ ⟨Barack Obama⟩ ⟨-⟩ ⟨Guantanamo Bay⟩

Table 7.1: Examples of topics correlations and dissenters

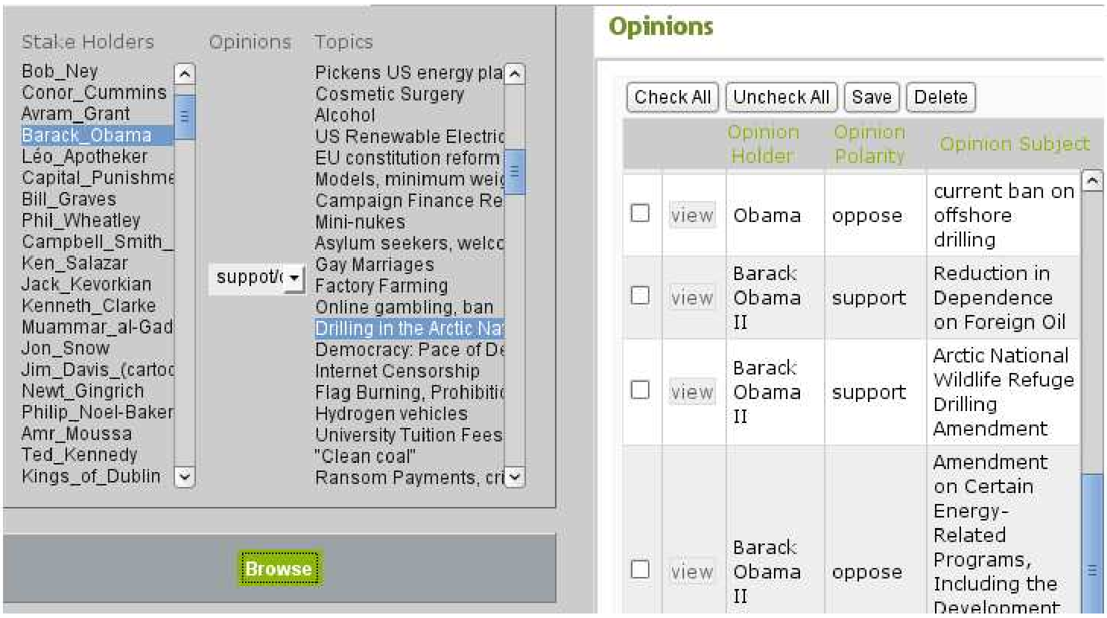


Figure 7.4: Flip-flop on "Offshore Drilling".

parties, or News media. Such kind of heatmaps can be used to analyze media bias. In cases where the stakeholders are political parties, or News media, we compute aggregated opinions polarity over all the opinions extracted from each news source or for each party on each topic. The result is a heatmap of opinions like the one in Figure 7.5. The heatmap in the figure is generated over the aggregated polarities of the opinions extracted from each news media (e.g. CNN, BBC, etc.) and on each topic (e.g. gun control, capital punishment, etc.).

7.5 Related Work

7.5.1 Contrastive Summarization

Bias and diversity of opinions have been studied in [80, 61, 96, 65, 45]. For example, [96] describes an approach for identifying major opponents (politicians or other named entities) in news corpora, and for classifying newly seen articles. However, this work does not consider the polarity of the article regarding fined grained topics. [72] identifies typical terms for a political party during different legislative periods from parliament speeches, thus revealing the different focuses of each party. They compare the politi-

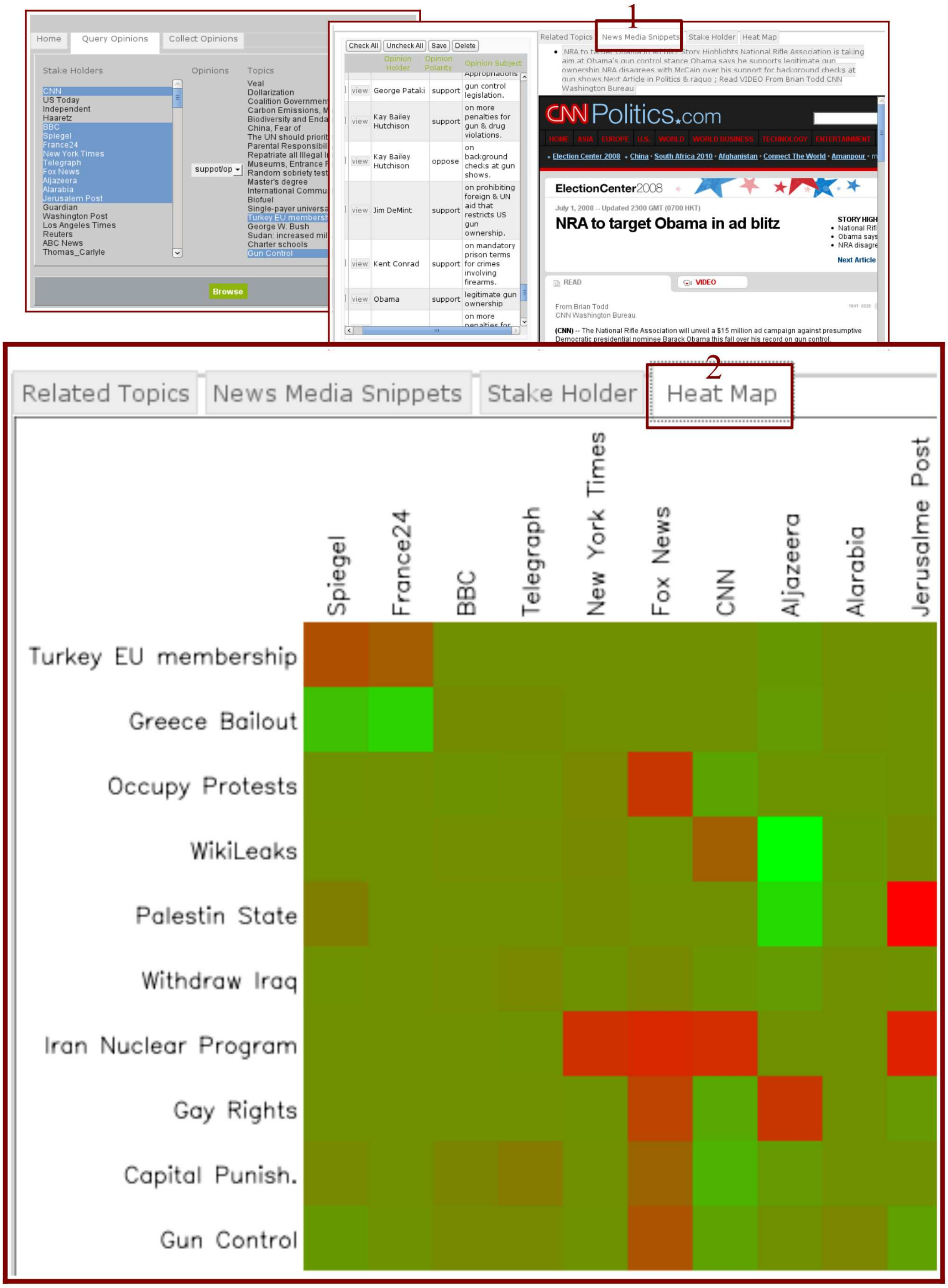


Figure 7.5: Heatmap of topics and media. Green, red, and yellow colors indicate positive, negative, and neutral opinions respectively

cal speeches with current news articles based on corpus statistics and used vocabulary. Other works seek to generate unbiased summaries of opinions from customer reviews [97, 68, 76, 14]. These methods are not suited for summarizing political controversies.

7.5.2 Bias and Diversity

Bias and diversity of opinions have been studied in [80, 61, 96, 65, 45]. For example, [96] describes an approach for identifying major opponents (politicians or other named entities) in news corpora, and for classifying newly seen articles. However, this work does not consider the polarity of the article regarding fined grained topics. [72] identifies typical terms for a political party during different legislative periods from parliament speeches, thus revealing the different focuses of each party. They compare the political speeches with current news articles based on corpus statistics and used vocabulary. Other works seek to generate unbiased summaries of opinions from customer reviews [97, 68, 76, 14]. These methods are not suited for summarizing political controversies.

7.5.3 Systems for Online Opinions

Several works explore online opinions. For example in [5] they present a system for social analytics on news. Their system interactively summarizes news articles and tweets comments around them. Related to this idea of understanding online opinions is described in [64]. Another work addressing the task of identifying controversial events using Twitter is described in [101].

Chapter 8

Conclusions

In this thesis, we described OpinioNetIt: a suite of methods and a full fledged system for building an opinion-base of facets, opinion holders and their opinions, and utilizes this opinion-base in applications that require political analysis. Our system acquired opinions and extracted facets from Web result snippets of online news sources using an initial set of seed patterns. These facets were then canonicalized and hierarchically organized. In addition, for each facet, an opinion holder was identified from the same snippets. Further opinions were collected using a lexicon of support/oppose phrases that was automatically built using a small set of opinions. Our evaluation showed an overall precision of 72%.

In addition and in order to populate the opinion-base with further opinions, we addressed the problem of automatically classifying opinion sentences about political debates which appear in news media and online forums, by politicians or other opinion makers, into fine-grained controversial topics and a pro/con polarity for each topic. We proposed a topic/polarity classification *approach that maps opinion sentences onto one or more topics in a category system of political debates, containing more than a thousand fine-grained topics. Our method builds on the estimation of statistical language models on a variety of advanced features. These features were specifically designed to overcome a major hurdle: the brevity of opinion sentences leading to sparseness of features. We showed the effectiveness of our techniques through systematic experiments. Our best method achieved a precision of about 74%, quite a positive result considering the hardness of the problem.*

In conclusion, not only are our techniques useful for building the opinion-base, and for automatically classifying the numerous utterances of politicians, but also serve as building blocks for sophisticated applications such as identifying flip-floppers and dis-

senters, and generating heat maps of political biases in the media.

There are several directions for future work. First, we identified generic facets (such as "the bill", "the protests", etc.) to be a major source of correct but unsatisfactory facts. If more specific facets are to be identified, then a closer look at the snippet, and possibly the article itself may be required. And we may need to use techniques from co-reference resolution to identify "which bill?" and "which protest?" Second, deep parsing techniques are sometimes limited when it comes to long sentences, resulting in either wrong or incomplete facet extractions. In order to improve the accuracy of our system, we either need to predict an inaccurate extraction or use additional heuristics to improve the extraction. Finally, we plan to expand our techniques to other kinds of sources. This is a big challenge because of the wide variety of writing styles. While the newspaper style is more factual and easier for opinion extraction, deploying similar techniques to blogs and online forums would result in much lower accuracy.

Bibliography

[1] *Lada Adamic and Natalie Glance. The Political Blogosphere and the 2004 US Election: Divided they Blog.* The International Workshop on Link Discovery (LinkKDD), *2005.*

[2] *Alina Adreevskaia and Sabine Bergler. Mining WordNet for Fuzzy Sentiment: Sentiment Tag Extraction from WordNet Glosses. In* the Conference of the European Chapter of the Association for Computational Linguistics (EACL), *2006.*

[3] *Eugene Agichtein and Luis Gravano. Snowball: Extracting Relations from Large Plain-Text Collections. In* the ACM International Conference on Digital Libraries (ICDL), *2000.*

[4] *Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. Subjectivity Word Sense Disambiguation. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP), *2009.*

[5] *Sihem Amer-Yahia, Samreen Anjum, Amira Ghenai, Aysha Siddique, Sofiane Abbar, Sam Madden, Adam Marcus, and Mohammed El-Haddad. MAQSA: A System for Social Analytics on News. In* the ACM International Conference on Management of Data (SIGMOD), *2012.*

[6] *Alina Andreevskaia and Sabine Bergler. Semantic Tagging at the Sense Level. In* the AAAI Conference on Artificial Intelligence (AAAI), *2006.*

[7] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. Language-Model-Based Pro/Con Classification of Political Text. In* the ACM International Conference on Information Retrieval (SIGIR), *2010.*

[8] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. OpinioNetIt: Understanding the Opinions-People Network for Politically Controversial Topics. In*

the ACM International Conference on Information and Knowledge Management (CIKM), *2011.*

[9] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. Harmony and Dissonance: Organizing the People's Voices on Political Controversies. In* the International Conference on Web Search and Data Mining (WSDM), *2012.*

[10] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. OpinioNetIt: A Structured and Faceted Knowledge-base of Opinions. In* the IEEE International Conference on Data Mining (ICDM), *2012.*

[11] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. Opinions Network for Politically Controversial Topics. In* the Workshop on Politics, Elections and Data (PLEAD)@CIKM, 2012, *2012.*

[12] *Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. PolariCQ: Polarity Classification of Political Quotations. In* the ACM International Conference on Information and Knowledge Management (CIKM), *2012.*

[13] *Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In* the International Conference on Language Resources and Evaluation (LREC), *2010.*

[14] *Alexandra Balahur, Ester Boldrini, Andrés Montoyo, and Patricio Martínez-Barco. A Unified Proposal for Factoid and Opinionated Question Answering. In* the International Conference on Computational Linguistics (COLING), *2010.*

[15] *Alexandra Balahur, Zornitsa Kozareva, and Andrés Montoyo. Determining the Polarity and Source of Opinions Expressed in Political Debates. In* The International Conference on Computational Linguistics and Intelligent Text Processing (CICLing), *2009.*

[16] *Alexandra Balahur, Ralf Steinberger, Erik van der Gootik, Bruno Pouliquen, and Mijail Kabadjov. Opinion Mining on Newspaper Quotations. In* The IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, *2009.*

[17] *Ramnath Balasubramanyan, William W. Cohen, Doug Pierce, and David P. Redlawsk. Modeling Polarizing Topics: When Do Different Political Communities Respond Differently to the Same News? In* the International AAAI Conference on Weblogs and Social Media (ICWSM), *2012.*

[18] *Carmen Banea, Rada Mihalcea, and Janyce Wiebe. A Bootstrapping Method for Building Subjectivity Lexicons for Languages with Scarce Resources. In* the International Conference on Language Resources and Evaluation (LREC), *2008.*

[19] *Mohit Bansal, Claire Cardie, and Lillian Lee. The Power of Negative Thinking: Exploiting Label Disagreement in the Min-cut Classification Framework. In* the International Conference on Computational Linguistics (COLING), *2008.*

[20] *Steven Bethard, Hong Yu, Ashley Thornton, Vasileios Hatzivassiloglou, and Dan Jurafsky. Automatic Extraction of Opinion Propositions and their Holders. In* In AAAI Spring Symposium on Exploring Attitude and Affect in Text, *2004.*

[21] *David Blei. Introduction to Probabilistic Topic Models.* Communications of the ACM, *2011.*

[22] *David Blei, Andrew Ng, and Michael Jordan. Latent Dirichlet Allocation.* Journal of Machine Learning Research, *3, 2003.*

[23] *John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, Bollywood, Boomboxes and Blenders: Domain Adaptation for Sentiment Classification. In* the Meeting of the Association of Computational Linguistics (ACL), *2007.*

[24] *Eric Breck, Yejin Choi, and Claire Cardie. Identifying Expressions of Opinion in Context. In* the International Joint Conference on Artifical iIntelligence (IJCAI), *2007.*

[25] *Sergey Brin. Extracting Patterns and Relations from the World Wide Web. In* The International Workshop on The World Wide Web and Databases, *1999.*

[26] *Clinton Burfoot, Steven Bird, and Timothy Baldwin. Collective Classification of Congressional Floor-Debate Transcripts. In* the Meeting of the Association of Computational Linguistics (ACL), *2011.*

[27] Giuseppe Carenini, Raymond Ng, and Ed Zwart. Extracting Knowledge from Evaluative Text. In the International Conference on Knowledge Capture (K-CAP)*, 2005.*

[28] Yejin Choi, Eric Breck, and Claire Cardie. Joint Extraction of Entities and Relations for Opinion Recognition. In the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2006.*

[29] Yejin Choi and Claire Cardie. Identifying Sources of Opinions with Conditional Random Fields and Extraction Patterns. In the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2005.*

[30] Philipp Cimiano, Andreas Hotho, and Steffen Staab. Learning Concept Hierarchies from Text Corpora Using Formal Concept Analysis. Journal of Artificial Intelligence Research*, 24(1), 2005.*

[31] Jacob Cohen. A coefficient of agreement for nominal scales. Educational and Psychological Measurement*, 20(1), 1960.*

[32] Michael Conover, Jacob Ratkiewicz, Matthew Francisco, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. Political Polarization on Twitter. In the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.*

[33] W. Bruce Croft and John Lafferty. Language Modeling for Information Retrieval. *Kluwer Academic Publishers, 2003.*

[34] Alex Davies and Zoubin Ghahramani. Language-independent Bayesian Sentiment Mining of Twitter. In The Workshop on Social Network Mining and Analysis (SNA-KDD)*, 2011.*

[35] Carrillo de Albornoz, Jorge, Plaza, Laura, and Pablo Gervás. Improving Emotional Intensity Classification using Word Sense Disambiguation. Special issue: Natural Language Processing and its Applications. Journal on Research in Computing Science*, 2010.*

[36] Xiaowen Ding, Bing Liu, and Philip Yu. A Holistic Lexicon-based Approach to Opinion Mining. In the International Conference on Web Search and Data Mining (WSDM)*, 2008.*

[37] *Anhai Doan, Luis Gravano, Raghu Ramakrishnan, and Shivakumar Vaithyanathan. Special Issue on Managing Information Extraction. In* the ACM International Conference on Management of Data (SIGMOD)*, 2008.*

[38] *Miles Efron. Cultural Orientation: Classifying Subjective Documents by Cociation Analysis. In* the AAAI Fall Symposium on Style and Meaning in Language, Art, Music, and Design*, 2004.*

[39] *Koji Eguchi and Victor Lavrenko. Sentiment Retrieval Using Generative Models. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2006.*

[40] *Andrea Esuli, Stefano Baccianella, and Fabrizio Sebastiani. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In* the International Conference on Language Resources and Evaluation (LREC)*, 2010.*

[41] *Andrea Esuli and Fabrizio Sebastiani. Determining the Semantic Orientation of Terms through Gloss Classification. In* the ACM International Conference on Information and Knowledge Management (CIKM)*, 2005.*

[42] *Andrea Esuli and Fabrizio Sebastiani. Determining Term Subjectivity and Term Orientation for Opinion Mining. In* the Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2006.*

[43] *Andrea Esuli and Fabrizio Sebastiani. PageRanking WordNet Synsets: An Application to Opinion Mining. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2007.*

[44] *Ana-Maria Popescuand Oren Etzioni. Extracting Product Features and Opinions from Reviews. In* the International Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT)*, 2005.*

[45] *Yi Fang, Luo Si, Naveen Somasundaram, and Zhengtao Yu. Mining Contrastive Opinions on Political Texts Using Cross-perspective Topic Model. In* the International Conference on Web Search and Data Mining (WSDM)*, 2012.*

[46] *Christiane Fellbaum.* WordNet: An Electronic Lexical Database. *MIT Press, 1998.*

[47] *Kavita Ganesan, ChengXiang Zhai, and Jiawei Han. Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions. In* the International Conference on Computational Linguistics (COLING)*, 2010.*

[48] *Namrata Godbole, Manjunath Srinivasaiah, and Steven Skiena. Large-Scale Sentiment Analysis for News and Blogs. In* the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2007.*

[49] *Honglei Guo, Huijia Zhu, Zhili Guo, XiaoXun Zhang, and Zhong Su. Product Feature Categorization with Mmultilevel Latent Semantic Association. In* the ACM International Conference on Information and Knowledge Management (CIKM)*, 2009.*

[50] *Trevor Hastie, Robert Tibshirani, and Jerome Friedman.* The Elements of Statistical Learning. *Springer, 2 edition, 2009.*

[51] *Vasileios Hatzivassiloglou and Kathleen R. McKeown. Predicting the Semantic Orientation of Adjectives. In* the Conference on European chapter of the Association for Computational Linguistics (EACL)*, 1997.*

[52] *Djoerd Hiemstra. A Linguistically Motivated Probabilistic Model of Information Retrieval. In* the European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, 1998.*

[53] *Djoerd Hiemstra.* Using Language Models for Information Retrieval. *PhD thesis, University of Twente, Enschede, 2001.*

[54] *HiroshiKanayama and Tetsuya Nasukawa. Fully Automatic Lexicon Expansion for Domain-oriented Sentiment Analysis. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2006.*

[55] *Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2011.*

[56] *Thomas Hofmann. Probabilistic Latent Semantic Indexing. In* the ACM International Conference on Information Retrieval (SIGIR)*, 1999.*

[57] *Minqing Hu and Bing Liu. Mining and Summarizing Customer Reviews. In* the International Conference on Knowledge Discovery and Data Mining (KDD)*, 2004.*

[58] *Matthew Hurst and Kamal Nigam. Retrieving Topical Sentiments from Online Document Collections. In* In Document Recognition and Retrieval (XI)*, 2004.*

[59] *Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. Target-Dependent Twitter Sentiment Classification. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2011.*

[60] *Thorsten Joachims.* Learning to Classify Text Using Support Vector Machines – Methods, Theory, and Algorithms*. Kluwer/Springer, 2002.*

[61] *Mouna Kacimi and Johann Gamper. Diversifying Search Results of Controversial Queries. In* the ACM International Conference on Information and Knowledge Management (CIKM)*, 2011.*

[62] *Rianne Kaptein, Maarten Marx, and Jaap Kamps. Who Said What to Whom?: Capturing the Structure of Debates. In* the ACM International Conference on Information Retrieval (SIGIR)*, 2009.*

[63] *George Karypis and Vipin Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs.* The SIAM Journal on Scientific Computing*, 20(1), 1998.*

[64] *Michael Kaschesky, Pawel Sobkowicz, and Guillaume Bouchard. Opinion Mining in Social Media: Modeling, Simulating, and Visualizing Political Opinion Formation in the Web. In* the International Conference on Digital Government Research (DG.O)*, 2011.*

[65] *Daisuke Kawahara, Kentaro Inui, and Sadao Kurohashi. Identifying Contradictory and Contrastive Relations Between Statements to Outline Web Information on a Given Topic. In* the International Conference on Computational Linguistics (COLING)*, 2010.*

[66] *Noriaki Kawamae. Joint sentiment aspect model. In* The Annual ACM Symposium on Applied Computing (SAC)*, 2012.*

[67] *Hyun Duk Kim, Kavita Ganesan, Parikshit Sondhi, and ChengXiang Zhai. Comprehensive Review Of Opinion Summarization. 2011.*

[68] *Hyun Duk Kim and ChengXiang Zhai. Generating Comparative Summaries of Contradictory Opinions in Text. In* the ACM International Conference on Information and Knowledge Management (CIKM)*, 2009.*

[69] *Soo-Min Kim and Eduard Hovy. Identifying and Analyzing Judgment Opinions. In* the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2006.*

[70] *Youngho Kim, Yoonjung Choi, and Sung-Hyon Myaeng. Generating Domain-Specific Clues Using News Corpus for Sentiment Classification. In* the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2010.*

[71] *Dan Klein and Christopher Manning. Accurate Unlexicalized Parsing. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2003.*

[72] *Ralf Krestel, Alex Wall, and Wolfgang Nejdl. Treehugger or Petrolhead?: Identifying Bias by Comparing Online News Articles with Political Speeches. In* the International Conference on World Wide Web (WWW)*, 2012.*

[73] *Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Opinion Extraction, Summarization and Tracking in News and Blog Corpora. In* AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.*

[74] *John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In* The International Conference on Machine Learning (ICML)*, 2001.*

[75] *Himabindu Lakkaraju, Chiranjib Bhattacharyya, Indrajit Bhattacharya, and Srujana Merugu. Exploiting Coherence for the Simultaneous Discovery of Latent Facets and Associated Sentiments. In* the SIAM International Conference on Data Mining (SDM)*, 2011.*

[76] *Kevin Lerman and Ryan McDonald. Contrastive Summarization: an Experiment with Consumer Reviews. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2009.*

[77] *Percy Liang, Slav Petrov, Michael Jordan, and Dan Klein. The infinite pcfg using hierarchical dirichlet processes. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2007.*

[78] *Chenghua Lin, Yulan He, and Richard Everson. A Comparative Study of Bayesian Models for Unsupervised Sentiment Detection. In* The Conference on Computational Natural Language Learning (CoNLL)*, 2010.*

[79] *Wei-Hao Lin and Alexander Hauptmann. Are These Documents Written from Different Perspectives?: A Test of Different Perspectives Based on Statistical Distribution Divergence. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2006.*

[80] *Yu-Ru Lin, James Bagrow, and David Lazer. More Voices Than Ever? Quantifying Media Bias in Networks. In* AAAI*, 2011.*

[81] *Bing Liu. Sentiment Analysis and Subjectivity. In* Handbook of Natural Language Processing, Second Edition*. 2010.*

[82] *Kun-Lin Liu, Wu-Jun Li, and Minyi Guo. Emoticon Smoothed Language Models for Twitter Sentiment Analysis. In* the AAAI Conference on Artificial Intelligence (AAAI)*, 2012.*

[83] *Avishay Livne, Matthew P. Simmons, Eytan Adar, and Lada A. Adamic. The Party is Over Here: Structure and Content in the 2010 Election. In* the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.*

[84] *Yue Lu, Huizhong Duan, Hongning Wang, and ChengXiang Zhai. Exploiting Structured Ontology to Organize Scattered Online Opinions. In* the International Conference on Computational Linguistics (COLING)*, 2010.*

[85] *Yue Lu, ChengXiang Zhai, and Neel Sundaresan. Rated aspect summarization of short comments. In* the International Conference on World Wide Web (WWW)*, 2009.*

[86] *Robert Malouf and Tony Mullen. Taking Sides: User Classification for Informal Online Political Discourse.* Internet Research*, 18, 2008.*

[87] *Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs. In* the International Conference on World Wide Web (WWW)*, 2007.*

[88] *David Miller, Tim Leek, and Richard Schwartz. A Hidden Markov Model Information Retrieval System. In* the ACM International Conference on Information Retrieval (SIGIR)*, 1999.*

[89] *Soo min Kim and Eduard Hovy. Crystal: Analyzing predictive opinions on the web. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2007.*

[90] *Arjun Mukherjee and Bing Liu. Modeling Review Comments. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2012.*

[91] *Dong Nguyen, Elijah Mayfield, and Carolyn Rosé. An Analysis of Perspectives in Interactive settings. In* The Workshop on Social Media Analytics (SOMA)*, 2010.*

[92] *Bo Pang and Lillian Lee. Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales, year = 2005, booktitle = acl.*

[93] *Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2004.*

[94] *Bo Pang and Lillian Lee. Opinion Mining and Sentiment Analysis.* Foundations and Trends in Information Retrieval*, 2(1–2), 2008.*

[95] *Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification Using Machine Learning Techniques. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2002.*

[96] *Souneil Park, KyungSoon Lee, and Junehwa Song. Contrasting Opposing Views of News Articles on Contentious Issues. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2011.*

[97] *Michael J. Paul, ChengXiang Zhai, and Roxana Girju. Summarizing Contrastive Viewpoints in Opinionated Text. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2010.*

[98] *Wei Peng and Dae Hoon Park. Generate Adjective Sentiment Dictionary for Social Media Sentiment Analysis Using Constrained Nonnegative Matrix Factorization. In* the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.*

[99] *Jay Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In* the ACM International Conference on Information Retrieval (SIGIR)*, 1998.*

[100] *Simone Paolo Ponzetto and Roberto Navigli. Knowledge-Rich Word Sense Disambiguation Rivaling Supervised Systems. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2010.*

[101] *Ana-Maria Popescu and Marco Pennacchiotti. Detecting Controversial Events from Twitter. In* the ACM International Conference on Information and Knowledge Management (CIKM)*, 2010.*

[102] *Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Expanding Domain Sentiment Lexicon through Double Propagation. In* The International Joint Conferences on Artificial Intelligence (IJCAI)*, 2009.*

[103] *Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Opinion Word Expansion and Target Extraction through Double Propagation.* Computational Linguistics*, 37(1), 2011.*

[104] *Balamurali A. R., Aditya Joshi, and Pushpak Bhattacharyya. Harnessing WordNet Senses for Supervised Sentiment Classification. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2011.*

[105] *Vassiliki Rentoumi, George Giannakopoulos, Vangelis Karkaletsis, and George Vouros. Sentiment Analysis of Figurative Language using a Word Sense Disambiguation Approach. In* The International Conference on Recent Advances in Natural Language Processing (RANLP)*, 2009.*

[106] *Ellen Riloff and Janyce Wiebe. Learning Extraction Patterns for Subjective Expressions. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2003.*

[107] *Mark Sanderson and Bruce Croft. Deriving Concept Hierarchies From Text. In* the ACM International Conference on Information Retrieval (SIGIR)*, 1999.*

[108] *Sunita Sarawagi. Information Extraction.* Foundations and Trends in Databases*, 1(3), 2008.*

[109] *Christina Sauper, Aria Haghighi, and Regina Barzilay. Content Models with Attitude. In* the Meeting of the Association of Computational Linguistics (ACL)*, 2011.*

[110] *Swapna Somasundaran and Janet Wiebe. Recognizing Stances in Ideological Online Debates. In* The NAACL-HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text (CAAGET)*, 2010.*

[111] *Swapna Somasundaran, Janyce Wiebe, and Josef Ruppenhofer. Discourse Level Opinion Interpretation. In* the International Conference on Computational Linguistics (COLING)*, 2008.*

[112] *Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2011.*

[113] *Mark Steyvers and Tom Griffiths.* Probabilistic Topic Models*. Lawrence Erlbaum Associates, 2007.*

[114] *Veselin Stoyanov, Claire Cardie, and Janyce Wiebe. Multi-perspective Question Answering Using the OpQA Corpus, booktitle = the International Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLTEMNLP), year = 2005.*

[115] *Qi Su, Xinying Xu, Honglei Guo, Zhili Guo, Xian Wu, Xiaoxun Zhang, Bin Swen, and Zhong Su. Hidden Sentiment Association in Chinese Web Opinion Mining. In* the International Conference on World Wide Web (WWW), *2008.*

[116] *Fabian Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: A Core of Semantic Knowledge. In* the International Conference on World Wide Web (WWW), *2007.*

[117] *Hiroya Takamura, Takashi Inui, and Manabu Okumura. Extracting Semantic Orientations of Phrases from Dictionary. In* the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL), *2007.*

[118] *Hristo Tanev. Unsupervised Learning of Social Networks from a Multiple-Source News Corpus. In* The Workshop of Multi-source, Multilingual Information Extraction and Summarization, *2007.*

[119] *Matt Thomas, Bo Pang, and Lillian Lee. Get out the Vote: Determining Support or Opposition from Congressional Floor-debate Transcripts. In* the International Conference on Empirical Methods on Natural Language Processing (EMNLP), *2006.*

[120] *Ivan Titov and Ryan McDonald. A Joint Model of Text and Aspect Ratings for Sentiment Summarization. In* the Meeting of the Association of Computational Linguistics (ACL), *2008.*

[121] *Ivan Titov and Ryan McDonald. Modeling Online Reviews with Multi-grain Topic Models. In* the International Conference on World Wide Web (WWW), *2008.*

[122] *Peter Turney. Thumbs Up or Thumbs Down?: Semantic Orientation Applied to Unsupervised Classification of Reviews. In* the Meeting of the Association of Computational Linguistics (ACL), *2002.*

[123] *Peter Turney and Michael Littman. Unsupervised Learning of Semantic Orientation from a Hundred-Billion-Word Corpus. Technical Report ERB-1094, National Research Council, Canada, Institute for Information Technology, 2002.*

[124] *Wen Wang, Sibel Yaman, Kristin Precoda1, Colleen Richey, and Geoffrey Raymond. Detection of Agreement and Disagreement in Broadcast Conversations. In* the Meeting of the Association of Computational Linguistics (ACL), *2011.*

[125] *Wei Wei and Jon Atle Gulla. Sentiment Learning on Product Reviews via Sentiment Ontology Tree. In* the Meeting of the Association of Computational Linguistics (ACL), *2010.*

[126] *Albert Weichselbraun, Stefan Gindl, and Arno Scharl. Using Games with a Purpose and Bootstrapping to Create Domain-specific Sentiment Lexicons. In* the ACM International Conference on Information and Knowledge Management (CIKM), *2011.*

[127] *Janyce Wiebe, Rebecca Bruce, and Thomas O'Hara. Development and Use of a Gold-standard Data Set for Subjectivity Classifications. In* the Meeting of the Association of Computational Linguistics (ACL), *1999.*

[128] *Janyce Wiebe, Theresa Wilson, and Matthew Bell. Identifying Collocations for Recognizing Opinions. In* the Workshop on Collocation: Computational Extraction, Analysis, and Exploitation, *2001.*

[129] *Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. Learning Subjective Language.* Computational Linguistics, *30(3).*

[130] *Edwin Wilson. Probable Inference, the Law of Succession, and Statistical Inference.* Journal of the American Statistical Association, *22(158), 1927.*

[131] *Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing Contextual Polarity in Phrase-level Sentiment Analysis.*

[132] *Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In* the IEEE International Conference on Data Mining (ICDM), *2003.*

[133] *Bei Yu, Stefan Kaufmann, and Daniel Diermeier. Classifying Party Affiliation from Political Speech.* Information Technology & Politics, *5, 2008.*

[134] Bei Yu, Stefan Kaufmann, and Daniel Diermeier. Exploring the Characteristics of Opinion Expressions for Political Opinion Classification. In the International Conference on Digital Government Research (DG.O)*, 2008.*

[135] Hong Yu and Vasileios Hatzivassiloglou. Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In the International Conference on Empirical Methods on Natural Language Processing (EMNLP)*, 2003.*

[136] ChengXiang Zhai. Statistical Language Models for Information Retrieval. Foundations and Trends in Information Retrieval*, 2(3), 2008.*

[137] Zhongwu Zhai, Bing Liu, Hua Xu, and Peifa Jia. Constrained LDA for Grouping Product Features in Opinion Mining. In the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, 2011.*

[138] Xiaodan Daniel Zhou, Paul Resnick, and Qiaozhu Mei. Classifying the Political Leaning of News Articles and Users from User Votes. In the International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2011.*

[139] Li Zhuang, Feng Jing, and Xiao-Yan Zhu. Movie Review Mining and Summarization. In the ACM International Conference on Information and Knowledge Management (CIKM)*, 2006.*

List of Figures

List of Tables