EXTRACTING OPINIONS AND EVENTS FROM TEXT: JOINT INFERENCE APPROACHES

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Bishan Yang

February 2016

EXTRACTING OPINIONS AND EVENTS FROM TEXT: JOINT INFERENCE APPROACHES

Bishan Yang, Ph.D.

Cornell University 2016

With the rapid growth of text data on the Web and on personal devices, there is an increasing need to automatically process text and unlock different types of information from it. Opinions and events are two important types of information that appear ubiquitously in text. One represents subjective information, concerning a person's attitudes, beliefs, sentiment, judgements and evaluations, and the other represents factual information concerning what happens in the real world. The ability to extract and interpret opinions and events is essential for many natural language processing (NLP) applications such as news summarization, open-domain question answering, social media analysis, and government document management.

While NLP has made great progress on information extraction tasks such as named entity recognition (entities like persons, organizations and locations) and named entity resolution (determining references of entities), much less progress has been made on the extraction of complex information such as opinions and events. Existing methods mostly extract individual components and attributes of opinions and events without accounting for their dependencies. Moreover, they often make phrase- or sentence-level predictions without considering the larger discourse context, such as a document or a conversation.

This dissertation presents models that address these two shortcomings. To capture the interdependencies among different information elements, we pro-

pose models that can perform joint inference across different but related extraction subtasks, including joint opinion entity extraction and relation extraction, and joint opinion segmentation and attribute classification. Extensive experiments show that joint inference yields significant improvements when compared to standard approaches that combine the subtasks in a pipeline, and achieves state-of-the-art performance on the extraction subtasks.

To facilitate global discourse understanding, we explore machine learning techniques that allow the integration of linguistic evidence at multiple levels of context — at the word, sentence, and document level — into coherent probabilistic models. Specifically, we develop a structured learning approach that can leverage intra- and inter-sentential cues in fine-grained sentiment analysis, and a Bayesian clustering model for event coreference resolution within a document and across documents. In both applications, we demonstrate the advantages of learning from multiple levels of contextual evidence.

BIOGRAPHICAL SKETCH

Bishan Yang was born in Hainan, China and grew up in a medical family. She developed interests in math at an early age and later pursued a degree in Computer Science at Peking University. Since her junior year of college, she became interested in data mining research and worked on several projects developing scalable data mining solutions for customer behavior prediction in telecommunication. She was also fascinated by text analysis problems, especially how to teach computers to mine patterns and knowledge from unstructured text.

After completing her bachelor's and master's degrees at Peking University, she came to Cornell to pursue a doctorate degree in the area of natural language processing and machine learning. During her PhD, she was dedicated to building a natural language processing system that can recognize and interpret information beyond simple facts in text. She has also spent several productive summers at eBay Research Labs, Google Research, and Microsoft Research, working on natural language understanding and knowledge representation learning problems.

To my parents, Jian Yang and Xiaodan Chen, and my partner, Igor Labutov,

for your unconditional love and support.

ACKNOWLEDGEMENTS

First and foremost, I am deeply grateful to my advisor Claire Cardie. Her visionary ideas, insightful suggestions, and careful mentorship have been instrumental to my academic development. I would not have accomplished this dissertation without the constant encouragement and endless support from Claire. She is also a role model to me, and I wish to always keep her as my mentor and my friend.

I am also very thankful to my committee members Johannes Gehrke and Peter Frazier. One of my favorite courses at Cornell was Johannes's advanced database systems class. I particularly enjoyed working with Johannes on the course project, and have been greatly inspired by his profound knowledge and creative thinking. I would like to express a special thanks to Peter for his invaluable input to my research. His clever ideas and rigorous research methodology have been critical in solving the technical challenges in parts of this dissertation.

I would like to thank my exceptional mentors who inspired and helped me during my research internships: Scott Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng at Microsoft Research; Rahul Gupta, Amarnag Subramanya, Nevena Lazic, and Fernando Pereira at Google Research; Nish Parikh, Gyanit Singh, and Neel Sundaresan at eBay. I also thank other amazing collaborators who contributed to my research experience: Janyce Wiebe, Lingjia Deng, and Yoonjung Choi from the University of Pittsburgh; Carmen Banea and Rada Mihalcea from the University of Michigan; Yanchuan Sim from Carnegie Mellon University; and Kai-Wei Chang from the University of Illinois at Urbana-Champaign.

I would also like to thank members of the Cornell NLP and MLDG groups for many helpful discussions. Special thanks to Cristian Danescu-Niculescu-

Mizil, Jack Hessel, Ozan Irsoy, Nikos Karampatziakis, Arzoo Katiyar, Lillian Lee, Moontae Lee, Joonsuk Park, Karthik Raman, Xanda Schofield, Adith Swaminathan, Chenhao Tan, Lu Wang, Ainur Yessenalina, and Jason Yosinski for helpful comments on my research work.

Also thanks to Jian-Tao Sun at Microsoft, who was the first to introduce me to research in machine learning for NLP. I was very fortunate to have the opportunity to work with him. And thanks to my undergraduate advisors Tengjiao Wang and Dongqing Yang at Peking University, who gave me a lot of support and freedom in pursuing my research passion.

I thank my friends in Ithaca, especially my best friend Taomo Zhou for always keeping me warm and cheerful during grad school, and Elisavet Kozyri for being my office mate for five years and having many joyful conversations with me about life and academia. I thank my wonderful parents for giving me the strength and inspiration to be who I am today, and my lovely brother, Guanhua, for being always so optimistic and caring. Last but not least, I thank my soulmate, Igor, for bringing so much love and joy into my life. I look forward to our future adventures together.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

Developing computer systems that can automatically extract knowledge from text is one of the long-term goals of artificial intelligence. The recent years have witnessed a rapid growth of text data — e.g., news, government reports, online reviews, scientific articles, emails, and the user-generated content of social media. This has resulted in an increasing need for efficient techniques for automatic text processing, especially fine-grained processing at or below the sentence level, to unlock the detailed information in natural language that is necessary for many high-level applications, such as business intelligence, government policy making and personal decision making.

Traditionally, the process of identifying information from sentences and converting it into a machine-readable form is considered *Information Extraction* (IE) (Cardie, 1997). Early IE systems focused on filling in pre-defined database fields from texts in restricted topics, for example, terrorist incidents and microchip fabrication, and relied on manually constructed lexicons and domain-specific extraction patterns (Cowie and Lehnert, 1996). More recent IE research uses statistical and machine learning methods that can be trained using human-annotated data and generalized to large text corpora (McCallum, 2005). They have been widely applied to a number of major IE tasks, in particular, extracting named entities (e.g., names of people, places, organizations), relations between a pair of named entities (e.g., person-X works-at organization-Y) and events from news articles. In general, the more complex the information structure, the

more difficult the task. For example, the accuracy of state-of-the-art named entity extractors has reached the 90% level, while the accuracy of event extractors is still at the 50% or 60% level at best. My research goal in this dissertation is to develop machine-learning-based methods for improving the extraction of information with complex structure.

1.1.1 Opinion and Event Extraction

In this dissertation, we focus on the extraction of opinions and events, both of which have complex semantic structure and can be expressed in a wide variety of linguistic forms. They also cover a broad range of the types of textual information that people are interested in — from factual to subjective information. Consider the following sentence as an example:

> Hillary Clinton offered a defense of Obamacare.

The sentence describes an event *Hillary Clinton defended Obamacare* and also conveys *Hillary Clinton*'s positive opinion towards *Obamacare*. We want to build computer algorithms that can automatically extract such information.

Despite evolving largely separately, previous research on automatic extraction of opinions (e.g., Wiebe et al. (2005), Choi et al. (2005), Breck et al. (2007a), Stoyanov and Cardie (2008)) and events (e.g., Ahn (2006), Chen and Ji (2009)) both aim to extract frame-semantic structures. Figure 1.1 shows an opinion frame and an event frame corresponding to the above sentence. An opinion frame includes three key components: *Opinion Expression* (*Trigger*) — a text span that indicates an opinion, *Holder* — the person or entity that is express-

Figure 1.1: An opinion frame and an event frame.

ing the opinion, and *Target* — the target or topic of the opinion, and attributes of opinions such as *Polarity* and *Intensity*. An event frame includes *Trigger* — a word or a phrase that describes an event action, *Agent* — who performs the action, *Target* — the target of the action, and other types of event arguments such as *Time* (when the event happens) and *Location* (where the event happens).

The extraction of opinion and event frames can be decomposed into three types of extraction subtasks:

1. **Entity extraction**: identification of *entity* — a text span that describes one of a predefined set of objects or concepts. The task involves finding the starting and ending boundaries of all text entity spans and assigning a class to each, e.g., identifying that "Hillary Clinton" is an opinion holder and that "offered a defense" is an opinion expression.
2. **Relation extraction**: identification of relations that are defined between two or more entities. The task involves associating entities involved in the same relation, e.g., associating an opinion expression with the holder(s) of the opinion and with the target(s) of the opinion.
3. **Attribute classification**: assigning a value to a property of a text entity. For example, the polarity of the opinion expression "offered a defense" is

positive.

These extraction subtasks share a lot of similarities with the traditional IE tasks but are more complex. For entity extraction, in IE, the entities are typically restrict to be named entities, whereas in opinion and event extraction, the entities can be expressions of opinions, actions, or topics, which are more irregular in linguistic forms. For relation extraction, IE mostly considers binary relations but not n-nary relations. In general, to extract opinion and event information, a system needs to account for complex language structure and contextual-dependent meanings. However, existing methods for opinion and event extraction have two common shortcomings:

1. **Ignoring interdependencies among extraction subtasks**. Standard approaches to opinion and event extraction address different extraction subtasks in isolation. The predictions for the subtasks are typically combined in a pipeline to produce a complete representation of information. For instance, for opinion extraction, opinion expressions are often extracted first and used as the input to extract opinion holders and targets, or, to determine the attributes of the opinion. Similarly, for event extraction, the event triggers are usually identified first and used as the input for extracting the event arguments. Such pipeline architecture suffers from error propagation; that is, the errors made in earlier stages cannot be corrected and will be propagated to later stages. For example, if an opinion expression is mistakenly missed, the error can never be recovered by the opinion holder and target extractors or attribute classifiers, even though they may have high confidence in correcting such an error.
2. **Lacking a discourse-level understanding of text**: While the interpretation

of opinions and events is highly contextually dependent, existing methods often make predictions at the phrase or sentence level without considering the larger discourse context. For instance, standard approaches to fine-grained sentiment classification treat a phrase or sentence as an independent text instance out of context. However, the sentiment interpretation may be highly contextually dependent. Take for instance the sentence "The performance is very predictable." Without context, the sentence can express either positive or negative sentiment. If "the performance" refers to business performance then it is likely positive but if it refers to actor performance in a movie then it is negative.

An important problem involved in discourse understanding is coreference resolution. There has been very little work on coreference resolution for opinions and events. Existing systems mostly employ pairwise clustering techniques that have been well-studied in entity coreference resolution (Stoyanov and Cardie, 2006; Chen et al., 2009). They largely rely on coreference predictions between a pair of textual mentions but cannot easily account for the global properties of the reference structure. Moreover, they mostly consider coreference within a document but do not account for coreference signals across documents.

This dissertation introduces models that address these shortcomings. The inspiration for these models comes from the way humans seem to interpret text — simultaneously considering multiple sources of low-level information (both syntactic and semantic) and aggregate information across different parts of the text. We believe that such unified interpretation — joint inference — is fundamental for natural language understanding. The models introduced in the following chapters aim to perform joint inference for accurate extraction and in-

terpretation of opinions and events expressed in text. We hope that these joint inference models can also be useful for the automatic extraction of other types of semantic information from text.

1.1.2 Joint Inference

We consider two types of joint inference:

(1) **Joint inference across different types of extraction subtasks**. Our assumption is that allowing information to flow among different but relevant extraction subtasks can improve predictions on individual subtasks. Therefore, we design joint objective functions that can optimize the predictions of different subtasks simultaneously.

Specifically, we tackle joint modeling of opinion entity extraction and relation extraction. The goal is to allow the decisions about the text spans of opinion entities and the relations between opinion entities to be made simultaneously so that the uncertainty of individual decisions may be reduced, and error propagation be prevented. To this end, we formulate a joint inference objective that simultaneously optimizes the opinion entity extraction probabilities (including the probabilities of identifying text spans of opinion expressions, holders, and targets) and the relation extraction probabilities (including the probabilities of linking opinion expressions to opinion holders, to opinion targets and to implicit arguments) subject to global consistency constraints.

We also address joint modeling of opinion expression extraction and attribute classification. The first one is typically formulated as a structured pre-

diction problem that deals with determining the text segment of an opinion expression. While the second one is usually treated as a text classification problem that deals with assigning a label to a given text input. Although they target different types of outputs, their decisions are highly correlated — the decisions on which words to include in an opinion expression can affect what attributes the opinion expression can have. For example, "extremely satisfied" expresses positive sentiment with high intensity while without "extremely" the intensity is not as high. To make joint decisions, we model the joint distribution over the decision variables for opinion expression segmentation and attribute classification. We explore two types of joint models: one performs joint inference both during learning and inference, and the other only during inference.

(2) **Joint inference over multiple levels of contextual evidence**. Here our goal is to improve text understanding tasks by leveraging contextual evidence at multiple levels — at the word, sentence, document level. We tackle two important tasks in opinion and event extraction: one is fine-grained sentiment classification, and the other is event coreference resolution.

Fine-grained sentiment classification is generally a challenging problem as the input text contains only a few words and little redundancy. Much effort has been made in developing techniques that can exploit cues effectively within a phrase or a sentence, for example, capturing word dependencies (Nakagawa et al., 2010) and compositional structures (Socher et al., 2013). However, much less work exploits discourse context for fine-grained sentiment classification. This is mainly because the discourse context may introduce structural complexity to the model and cause impractical computational costs. To address this issue, we develop a structured learning approach that allows the modeling of discourse

context during learning and inference without introducing much computational cost.

Event coreference resolution is typically considered as a clustering problem, where the objects being clustered are textual mentions that describe events. Existing approaches mostly employ pairwise clustering techniques, which groups objects based on the coreference relations between pairs of objects. Although the pairwise coreference signals are important, they may not be strong enough to infer the global reference structure. There have been some attempts on Bayesian modeling of the global reference distribution. However, they cannot capture any prior information about pairwise coreference relations in the data. To address this problem, we propose a Bayesian clustering model that allows the incorporation of pairwise coreference signals as informative priors to guide the inference of the global cluster distribution.

1.2 Contributions

The primary contribution of this dissertation is the development of effective joint inference models for extracting semantic representations of opinions and events from natural language text. More specifically, we make the following contributions:

- **Joint opinion entity and relation extraction**. We propose a joint inference model that can extract opinion entities and their relations simultaneously from text. It is the first model that allows the joint extraction of all three types of opinion entities: opinion expressions, holders, and targets, and

accounts for opinion holders/targets that are not explicitly mentioned in text. We show that our model can significantly reduce errors in all the extraction subtasks compared to existing pipeline and joint approaches. This work is described in Chapter 3. We also mentioned its application to the extraction of event triggers, event arguments and their relations in Chapter 6.

- **Joint opinion segmentation and attribute classification**. We propose a semi-Markov CRF-based model for opinion expression extraction, which can make extraction decisions at the segment level rather than the word level, and extensions of the model to allow joint modeling of opinion segmentation and attribute classification. We define the joint distribution over the assignments of opinion expression segmentation and attribute values based on opinion segmentation and attribute specific potential functions. We explore two types of joint models: one estimates the segmentation- and attribute-specific parameters jointly while the other estimates them separately and combines them only during inference time. We show that the second model is more effective and efficient for the task. We also provide insights into the advantages of both types of joint models. This work is the subject of Chapter 4.

- **Context-aware sentiment analysis**. We propose a CRF-based approach to sentence-level sentiment classification, which can leverage both intra- and inter-sentential contextual cues by imposing soft structural constraints on the CRF posterior. We examine our model by comparing it to models that incorporate inter-sentential cues as hard constraints during inference, and we found that our model can utilize the contextual cues more effectively and collectively for sentence-level sentiment classification. Furthermore,

we show that our model can provide performance improvements in a semi-supervised learning setting where unlabeled data is used as distant supervision to assist learning. This work is presented in Chapter 5.

- **Within- and cross-document event coreference resolution**. We propose a Bayesian clustering model for within- and cross-document event coreference resolution. Unlike traditional Bayesian clustering models, our model allows the incorporation of pairwise coreference evidence as feature-rich priors for the inference of the cluster distribution. We show that our model yields substantial improvements over the state-of-the-art event coreference approaches on both within- and cross-document event coreference resolution. Moreover, our model is generally applicable to cluster any groups of objects that exhibit rich pairwise compatibility properties. This work is described in Chapter 6.

1.3 Roadmap

The rest of the dissertation is organized as follows. In Chapter 2, we present a more detailed description of the related work. In Chapter 3, we present a joint inference framework for opinion entity and relation extraction. In Chapter 4, we propose joint modeling techniques for combining opinion expression extraction with opinion attribute classification. In Chapter 5, we propose a context-aware model that can leverage intra- and inter-sentential cues to make more accurate sentiment predictions. Finally, in Chapter 6, we study the problem of coreference resolution of events. We apply our opinion extraction methods to the extraction of event mentions from text, and introduce a novel Bayesian clustering model for inferring within- and cross-document event coreference clusters.

CHAPTER 2

BACKGROUND AND RELATED WORK

In this chapter, we present an overview of the background and existing research in the area of opinion extraction and event extraction, as well as existing techniques for joint inference modeling that are related to the work presented in this dissertation. We further discuss the related work in the context of specific opinion and event extraction problems in the corresponding chapters.

2.1 Opinion Extraction

Research on analyzing opinions in text dates back to the mid 1990s. Wiebe (1994) was one of the first who recognized the importance of detecting subjectivity in sentences and proposed a computational approach to identify subjective sentences in third-person fictional narrative text. Hatzivassiloglou and McKeown (1997) later proposed the use of statistical techniques to predict the semantic orientations of conjoined adjectives. Wiebe et al. (1999) also applied statistical techniques to classify subjective sentences in news articles. Due to the lack of a sufficient amount of labeled data and the limit of the state of the art machine learning techniques, the systems developed at the time targeted relatively restricted classification tasks, and were trained and evaluated on small datasets.

Since the early 2000s, the proliferation of the Web provides researchers access to large collections of online opinion resources, such as product review sites, political forums, and personal blogs, in the meantime, it intensifies the needs for systems that can help people process information from a large amount of opin-

ionated documents online. Opinion analysis research, as a result, has received a surge of research interest (see Pang and Lee (2008) and Liu (2012) for in-depth surveys). A lot of research effort has been invested in analyzing opinions in online reviews, for example, classifying the polarity of a movie (or product) review to be positive or negative (Pang et al., 2002; Turney, 2002; Pang and Lee, 2004; Blitzer et al., 2007), identifying product features (or aspects) and predicting sentiment for these features (or aspects) (Hu and Liu, 2004a; Popescu and Etzioni, 2005; Ding et al., 2008), and predict the rating of a review on a certain scale (Pang and Lee, 2005; Goldberg and Zhu, 2006; Snyder and Barzilay, 2007). Research in this area is usually domain-dependent as reviews have certain properties that do not generalize to other types of text. For instance, the topics in reviews are generally restricted (e.g., a movie or a digital product) and known in advance. There is more regularity in language, e.g., opinions are usually explicitly expressed by sentiment-bearing adjectives, and product features are usually nouns. Such knowledge often contributes to the high performance of the models.

In this dissertation, we aim to develop domain-independent solutions to opinion analysis in general text documents. We focus on fine-grained opinion extraction, which deals with identifying and interpreting opinions at or below sentence level. Because normally not all sentences in a document express opinions, and the opinionated sentences may express opinions from different sources and towards different topics. In the following, we describe previous work on fine-grained opinion extraction in general and their relations to this dissertation.

2.1.1 Domain-independent Fine-grained Opinion Extraction

Earlier research that studied opinion extraction in general mainly focused on detecting subjective sentences or phrases (Wiebe et al., 1999; Wiebe et al., 2001; Riloff et al., 2003). In 2003, researchers began to discuss other aspects that are necessary for analyzing opinions in text (Cardie et al., 2003; Yu and Hatzivassiloglou, 2003), including determining who holds the opinion, what the opinion is about, and what are the attributes of the opinion (e.g., polarity and intensity). Knowing this information is important for question answering systems that target complex questions like "What was the public reaction to Hillary Clinton's 2016 presidential campaign announcement?" which require finding and organizing opinions in documents from multiple sources. Cardie et al. (2003) were the first to view the extraction of such information as an opinion-oriented information extraction task, and use opinion-oriented "scenario templates", which are similar to event-oriented scenario templates used in event-based information extraction, as a summary representation of opinions in text. However, they only proposed approaches for the task without empirical evaluation. In the subsequent years, great progress has been made on the development of techniques for different subproblems of fine-grained opinion extraction.

Opinion Expression Extraction. Breck et al. (2007b) was the first to apply sequence labeling models, in particular, Conditional Random Fields (CRFs) (Lafferty et al., 2001) to identify expressions that express opinions in sentences, for example, "very concerned", "strongly criticized". To employ CRFs, the expression-level opinion annotations are transformed into word-level annotations using BIO or IO encoding: each word is tagged as either *Beginning* an entity, being *Inside* an entity, or being *Outside* an entity. The decisions on whether

to include a word in an opinion expression count only on word-level signals, such as part-of-speech tags and WordNet categories. To account for relations between words, Johansson and Moschitti (2010b; 2010a; 2013a) developed several reranking approaches that can incorporate word relations derived from syntactic and semantic role information as features in a reranker for opinion expression extraction. However, the reranker is still relied on the outputs of a word-level sequence labeler. In Chapter 4, we develop a semi-Markov CRF based model that can perform segment-level sequence tagging based on information about phrases.

Opinion Holder Extraction. Bethard et al. (2004) first presented a statistical approach to identify sources of propositional opinions, which are opinions contained in sentential complements of verbs that introduce opinions, e.g., "believe", "criticize", "argue". Kim and Hovy (2004) and Choi et al. (2005) considered the extraction of opinion holders for opinions in general. Choi et al. (2006) considered extracting opinion expressions and opinion holders jointly by performing joint inference over separately-trained models. Johansson and Moschitti (2010a) applies reranking techniques to incorporate global features to better extract opinion expressions and opinion holders. Our work in Chapter 3 is the closest to the work of Choi et al. (2006) but differs in that it can jointly extract not only opinion expressions and holders but also opinion targets and can account for implicit holders and targets.

Opinion Target Extraction. As mentioned before, many techniques were developed in the context of product reviews where opinion targets refer to the features or aspects of specific products. However, very limited work has been done on opinion target extraction for general text documents. Stoyanov and Cardie

(2008) proposed several approaches for the automatic identification of target spans in newswire documents. However, their methods were based on simple syntactic rules and involved no learning components. In our work, we identify target spans by training a sequence labeling model using the target annotation provided by (Wilson, 2008b).

Fine-grained Sentiment Classification. The task of determining the sentiment expressed in a given text snippet — a phrase, clause or sentence — has received increasing attention in recent years. Standard text classification techniques such as Naive Bayes and maximum entropy classifiers make strong independence assumptions about the language structure and thus does not perform well on the task. Many techniques have been developed to account for the linguistic structure of language in classification models in order to better capture the meaning of opinions. Most such techniques focus on exploiting word relations within a sentence, for example, negators and intensifiers (Wilson et al., 2009; Ikeda et al., 2010), compositional structure of phrases (Yessenalina and Cardie, 2011; Choi and Cardie, 2008; Socher et al., 2013), and syntactic structure of sentences (Nakagawa et al., 2010). Very limited work has taken into account contextual relations among sentences. This is, in part, because conventional classification models cannot easily incorporate information about long-distance relationships. Existing work that incorporates discourse-level information either encode it as heuristic rules or hard constraints to the output of the sentiment classifier. For example, Taboada et al. (2008) assigned higher weight to the sentiment-bearing words in nuclei (the most relevant sentences in text, as defined in the Rhetorical Structure Theory (Mann and Thompson, 1988; Taboada and Mann, 2006)) in the calculation of semantic orientation. Somasundaran et al. (2009) used opinion target coreference annotation to impose hard constraints

on the polarity of sentences during inference. In this dissertation, we developed a model that allows the incorporation of discourse-level information as soft constraints during learning and inference of sentence-level sentiment.

2.2 Event Extraction

Early information extraction (IE) systems were event-based, focusing on extracting events from news documents about one particular topic. Message Understanding Conference (MUC) first created several extraction tasks, centering around filling a predefined "scenario template" with information extracted from text (Grishman and Sundheim, 1996). The template was defined for a particular topic, for example, a terrorist attack include the description of the attack itself, the people that were killed, and the facilities that were damaged. Approaches developed for such tasks relied on hand-coded task-specific patterns, and were evaluated on a small set of documents (see (Grishman, 2011) for a more detailed review).

Succeeding MUC, Automatic Content Extraction (ACE) provided a larger collection of news documents on a broader range of topics. ACE 2005[1] defined an event extraction task that deals with event types that frequently occur in news articles, like life events, business events, and conflict events. The extraction of an event includes the extraction of the event trigger — the main word (usually a verb or a noun) that describes the event, the assignment of event attributes (e.g., modality, polarity) — the identification of event arguments with roles (each event type has its own set of roles) and determining event corefer-

[1] http://www.itl.nist.gov/iad/mig/tests/ace/2005/doc/ace05-evalplan.v2a.pdf

ence within a document. Existing systems usually solve the extraction of event triggers and event arguments by pipelining multiple standard text classification models such as maximum-entropy-based classifiers. For example, Ahn (2006) built a binary classifier to identify event trigger words, a multi-class classifier to identify argument mentions (entity/time/value mentions) given an event trigger, and a multi-class classifier to assign event attributes. Ji and Grishman (2008) later used global statistics of event triggers and arguments in a document or document cluster to further improve the classification accuracy. In our work, we show how to adapt the joint inference approaches for opinion expression and opinion argument extraction to the extraction of event triggers and event arguments.

For event coreference, existing methods mostly employ models that have been well-studied for entity coreference resolution. Ahn (2006) used a mention-pair entity coreference model to identify the coreferential relations between event mentions. Similarly, Chen et al. (2009) applied a pairwise clustering algorithm by using event-specific features, derived from event triggers, arguments and attributes (polarity and modality). Recently, Bejan and Harabagiu (2010) developed a class of nonparametric Bayesian models for event coreference resolution. They treat event mentions as structured linguistic objects and infer the event clustering distributions based on the observations of the event mentions without any supervision. Bejan and Harabagiu (2010) also created the EventCorefBank (ECB) corpus, a newswire corpus that covers more diverse event types than the ACE corpus and provides within- and cross-document event coreference annotation. Lee et al. (2012) extended the ECB corpus by adding entity coreference annotation, and proposed a joint event and entity coreference resolution algorithm that iteratively merges verbal and nominal clusters

based on a linear regression model. Liu et al. (2014) have shown that propagating information alternatively between events and their arguments after conventional pairwise clustering can improve performance for within-document event coreference. Cybulska and Vossen (2014b) further extended the ECB corpus by adding event argument and argument type annotations as well as adding more news documents in order to increase the lexical diversity in the previous corpora. Our work uses the latest ECB corpus and develops a Bayesian clustering model that can leverage supervisory information about pairwise coreference relations while performing Bayesian inference for both within- and cross-document event coreference resolution.

Event extraction has also been studied in other types of text data like biomedical text (e.g., Poon and Vanderwende (2010)), social media messages (e.g., Ritter et al. (2012)), and biology textbooks (e.g., Berant et al. (2014)), where the main extraction task is to identify event-related entities and relations, without further aggregating them using coreference resolution.

2.3 Joint Inference Approaches

Considerable research effort has been invested on developing models that can perform joint inference across multiple steps of a text processing pipeline in order to reduce error propagation. Roth and Yih (2004; 2007) first introduced the idea of joint inference with multiple classifiers. They presented an integer linear programming (ILP) based approach that seeks a globally-optimal solution given the outputs of independently-trained classifiers and consistency constraints. In particular, they applied the approach to named entity type classification and

relation classification. Later work (Punyakanok et al., 2004; Punyakanok et al., 2008) applied the idea to semantic role labeling (SRL), the task of identifying argument spans and assigning argument types for each predicate in a given input sentence, where joint inference was performed over classifiers that were built independently for argument candidate identification and argument type classification. The ILP-based joint inference formulation has also been employed in other tasks like dependency parsing (Riedel and Clarke, 2006; Martins et al., 2009), verb SRL and preposition role labeling (Srikumar and Roth, 2011), opinion expression and holder extraction (Choi et al., 2006), and event extraction (Berant et al., 2014). Our work also employs the ILP-based joint inference. It is very closed to the work of (Choi et al., 2006). The difference is that Choi et al. (2006) only considered opinion expressions and holders, while our ILP formulation allows the joint extraction of opinion expressions, holders, and targets, and can account for implicit holders or targets that are not explicitly mentioned in text.

Joint inference can also be implemented using dual decomposition (Rush et al., 2010), which combines separately-trained models using linear programming and solves the relaxation problem instead of the original problem in order to speed up inference. Dual decomposition and its variants have been applied to syntactic parsing (Rush et al., 2010), machine translation (Chang and Collins, 2011), semantic role labeling (Das et al., 2012), and event extraction (Reichart and Barzilay, 2012). One downside of dual decomposition is that it does not always provides the optimal solution. Martins et al. (2011) proposed an AD^3 algorithm that can find the exact solution of the original ILP, however, it can have exponential runtime in the worst-case scenario.

Instead of combining separately-trained models during inference, we can

also train a joint model of multiple tasks. There has been much interest in formulating joint inference using probabilistic graphical models. For example, Finkel et al. (2006) proposed a conditional random field-based model for joint parsing and named entity recognition. Poon and Domingos (2007) proposed a Markov Logic Network for joint citation segmentation and matching. Singh et al. (2013) presented a factor graph that models the joint distribution over entity tags, relations between entities, and coreference decisions. Modeling the joint probability distribution allows information to flow across tasks both during learning and inference. However, they are usually very costly to train and require sufficient amount of labeled data that is jointly annotated with different types of task labels. In Chapter 4, we also develop several joint probability models and investigate the trade-off between joint learning and joint inference.

In this dissertation, we also investigate joint inference approaches for context-aware text understanding. The goal is to learn from multiple levels of contextual evidence by integrating them into probabilistic models. To this end, we explore the use of two modern machine learning techniques: one is posterior regularization, and the other is distance-dependent Chinese Restaurant Process.

Posterior regularization (PR) is a probabilistic framework that allows easy incorporation of prior knowledge as soft constraints on model posteriors (Ganchev et al., 2010). It has been successfully applied to many NLP tasks such as part-of-speech tagging and dependency parsing (Ganchev et al., 2010). Existing work explored structural constraints that encourage sparsity (Ganchev et al., 2010), structure agreements (Ganchev et al., 2009), and label existence (Bellare et al., 2009). In Chapter 5, we design several types of structural constraints based on intra- and inter-sentential discourse relations. Unlike previous work

that only incorporates the PR constraints during training, we incorporate the PR constraints during both training and inference.

Distance-dependent Chinese Restaurant Process (DDCRP) is a clustering framework that allows easy incorporation of prior knowledge about data dependencies as Bayesian priors (Blei and Frazier, 2011). It has been successfully used to account for data dependencies in many clustering tasks, for example, accounting for temporal relations between documents in topic modeling (Kim and Oh, 2011), document-pair similarities in document clustering (Socher et al., 2011a), morphological similarities between words in part-of-speech induction (Sirts et al., 2014), and spatial distance between pixels in image segmentation (Ghosh et al., 2011). In Chapter 6, we develop a hierarchical extension of the DDCRP for event coreference resolution. Unlike existing Bayesian clustering approaches for the task, it can encode different types of pairwise compatibility signals as feature-rich priors, and perform both within-document and cross-document clustering with such priors.

CHAPTER 3

JOINT OPINION ENTITY AND RELATION EXTRACTION

In this chapter, we present a joint inference framework for extracting opinion entities: *opinion expressions*, *opinion holders*, and *opinion targets* and opinion relations: *is-from* and *is-about* relations. The work described in this chapter is based on Yang and Cardie (2013).

As discussed in Chapter 1, while much progress has been made on the extraction of opinion expressions, opinion holders and targets (Choi et al., 2006; Kim and Hovy, 2006; Breck et al., 2007a; Kobayashi et al., 2007; Johansson and Moschitti, 2010b), most existing work addressed the extraction of these opinion-related entities in isolation. To construct a comprehensive opinion information representation, existing systems usually combine the different extraction components in a pipeline fashion. As a result, errors can easily propagate and accumulate throughout the pipeline. One exception is Choi et al. (2006), which proposed a joint approach to identify opinion holders, opinion expressions, and their association relations. Their approach, however, does not consider opinion targets nor does it allow opinion expressions to have missing holders or targets.

In this work, we present a joint inference approach that allows joint extraction of more than two types of opinion-related entities, in particular, *opinion expressions*, *opinion holders* and *opinion targets*, and their association relations, while taking into account missing arguments for each opinion expression (e.g., the opinion holder or target may not be explicitly expressed in text.) Consider the following examples in which opinion expressions (O) are underlined, and targets (T) and holders (H) of the opinion are bracketed.

S1: [The workers]$_{[H_{1,2}]}$ were irked$_{[O_1]}$ by [the government report]$_{[T_1]}$ and were worried$_{[O_2]}$ as they went about their daily chores.

S2: From the very start it could be predicted$_{[O_1]}$ that on the subject of economic globalization, [the developed states]$_{[T_{1,2}]}$ were going to come across fierce opposition$_{[O_2]}$.

The numeric subscripts denote *association relations*, one of IS-ABOUT or IS-FROM. In S1, for instance, opinion expression "were irked" (O_1) IS-ABOUT "the government report" (T_1). Note that the IS-ABOUT relation can contain an empty target (e.g., "were worried" in S1); similarly for IS-FROM w.r.t. the opinion holder (e.g., "predicted" in S2). We also allow an opinion entity to be involved in multiple relations (e.g., "the developed states" in S2).

We model opinion entity extraction as a sequence tagging problem and opinion relation extraction as a binary classification problem. We then propose a joint inference framework to jointly optimize the predictors for different subproblems with global consistency constraints. We hypothesize that joint inference can reduce the ambiguity in different extraction subproblems and thus, performance increased. For example, uncertainty w.r.t. the spans of opinion entities can adversely affect the prediction of opinion relations; and evidence of opinion relations might provide clues to guide the accurate extraction of opinion entities.

We evaluate our approach using a standard corpus for fine-grained opinion analysis (the MPQA corpus (Wiebe et al., 2005)) and demonstrate that our model outperforms by a significant margin traditional baselines that do not employ joint inference for extracting opinion entities and different types of opinion

relations.

3.1 Related Work

Significant research effort has been invested into fine-grained opinion extraction for open-domain text such as news articles (Wiebe et al., 2005; Wilson et al., 2009). Many techniques were proposed to identify the text spans for opinion expressions (Breck et al., 2007a; Johansson and Moschitti, 2010b; Johansson and Moschitti, 2010a), opinion holders (Choi et al., 2005; Kim and Hovy, 2006) and opinion targets (Stoyanov and Cardie, 2008). They usually extracted opinion holders and targets based on their relations to opinion expressions. For instance, Kim and Hovy (2006) identifies opinion holders and targets by using their semantic roles related to opinion words. Kobayashi et al. (2007) extracts "aspect-evaluation" relations (relations between opinion expressions and targets) by performing binary classification on target candidates given each opinion expression. Johansson and Moschitti (2010a) extract opinion holders by reranking the outputs of a sequence labeler using features constructed based on the syntactic and semantic relations between holder candidates and the previously-extracted opinion expressions. All these methods extract opinion expressions and opinion arguments (holders or targets) in separate stages instead of extracting them jointly.

Most similar to our method is Choi et al. (2006), which jointly extracts opinion expressions, holders and their IS-FROM relations using an ILP-based approach. In contrast, our approach (1) also considers the IS-ABOUT relation which is arguably more complex due to the larger variety in the syntactic structure ex-

hibited by opinion expressions and their targets, (2) handles implicit opinion relations (opinion expressions without any associated argument), and (3) uses a simpler ILP formulation.

There has also been substantial interest in opinion extraction from product reviews (Liu, 2012). Most existing approaches focus on the extraction of opinion targets and their associated opinion expressions and usually employ a pipeline architecture: identify opinion expressions first, and then use rule-based or machine-learning-based approaches to identify potential opinion targets (Hu and Liu, 2004a; Wu et al., 2009; Liu et al., 2012). In addition to pipeline approaches, bootstrapping-based approaches were proposed (Qiu et al., 2009; Qiu et al., 2011; Zhang et al., 2010) to identify opinion expressions and targets iteratively; however, they suffer from the problem of error propagation.

There is much work demonstrating the benefit of performing global inference. Roth and Yih (2004) proposed a global inference approach in the formulation of a linear program (LP) and applied it to the task of classifying named entities and relations simultaneously. Their problem is different from ours — it assumes that the text spans of named entities are known a priori, and only the class labels need to be assigned. Joint inference has also been applied to semantic role labeling (SRL) (Punyakanok et al., 2008; Srikumar and Roth, 2011; Das et al., 2012), where the focus is on jointly identifying different semantic arguments and their roles for a given predicate. The problem is conceptually similar to identifying opinion arguments for opinion expressions, however, in our problem, we want to extract opinion expressions along with their arguments rather than only extracting the arguments.

3.2 A Joint Inference Approach

In this section, we present our joint inference approach to opinion entity and relation extraction. We first describe how we model opinion entity extraction and opinion relation extraction separately, and then describe how we combine the separately-trained models in a joint inference framework using integer linear programming.

3.2.1 Opinion Entity Extraction

We formulate the task of opinion entity extraction as a sequence labeling problem and employ conditional random fields (CRFs) (Lafferty et al., 2001) to learn the probability of a sequence assignment $\mathbf{y}$ for a given sentence $\mathbf{x}$. Through inference, we can find the best sequence assignment for sentence x and recover the opinion entities according to the standard "BIO" encoding scheme. We consider three entity types: D, T, H, where D denotes opinion expressions, T denotes opinion targets, H denotes opinion holders.

We define potential function f_{iz} that gives the probability of assigning a span i with entity label z, and the probability is estimated based on the learned parameters from CRFs. Formally, given a within-sentence span $i = (a, b)$, where a is the starting position and b is the end position, and label $z \in \{D, T, H\}$, we have

$$f_{iz} = p(\mathbf{y}_a = B_z, \mathbf{y}_{a+1} = I_z, ..., \mathbf{y}_b = I_z, \mathbf{y}_{b+1} \neq I_z | \mathbf{x})$$

$$f_{iO} = p(\mathbf{y}_a = O, ..., \mathbf{y}_b = O | \mathbf{x})$$

These probabilities can be efficiently computed using the forward-backward

algorithm.

3.2.2 Opinion Relation Extraction

We consider extracting the IS-ABOUT and IS-FROM opinion relations. In the following we will not distinguish these two relations, since they can both be characterized as relations between opinion expressions and opinion arguments, and the methods for relation extraction are the same.

We treat the relation extraction problem as a combination of two binary classification problems: *opinion-arg classification*, which decides whether a pair consisting of an opinion candidate o and an argument candidate a forms a relation; and *opinion-implicit-arg classification*, which decides whether an opinion candidate o is linked to an implicit argument, i.e. no argument is mentioned. We define a potential function r to capture the strength of association between an opinion candidate o and an argument candidate a,

$$r_{oa} = p(y = 1|x) - p(y = 0|x)$$

where $p(y = 1|x)$ and $p(y = 0|x)$ are the logistic regression estimates of the positive and negative relations. Similarly, we define potential $r_{o\emptyset}$ to denote the confidence of predicting opinion span o associated with an implicit argument.

Opinion-Arg Relations

For *opinion-arg* classification, we construct candidates of opinion expressions and opinion arguments and consider each pair of an opinion candidate and an argument candidate as a potential opinion relation. Conceptually, all possible

subsequences in the sentence are candidates. To filter out candidates that are less reasonable, we consider the opinion expressions and arguments obtained from the n-best predictions by CRFs[1]. We also employ syntactic patterns from dependency trees to generate candidates. Specifically, we selected the most common patterns of the shortest dependency paths[2] between an opinion candidate o and an argument candidate a in our dataset, and include all pairs of candidates that satisfy at least one dependency pattern. For the IS-ABOUT relation, the top three patterns are (1) $o \uparrow_{dobj} a$, (2) $o \uparrow_{ccomp} x \uparrow_{nsubj} a$ (x is a word in the path that is not covered by either o nor a), (3) $o \uparrow_{ccomp} a$; for the IS-FROM relation, the top three patterns are (1) $o \uparrow_{nsubj} a$, (2) $o \uparrow_{poss} a$, (3) $o \downarrow_{ccomp} x \uparrow_{nsubj} a$.

Note that generating candidates this way will give us a large number of negative examples. Similar to the preprocessing approach in (Choi et al., 2006), we filter pairs of opinion and argument candidates that do not overlap with any gold standard relation in our training data.

Many features we use are common features in the SRL tasks (Punyakanok et al., 2008) due to the similarity of opinion relations to the predicate-argument relations in SRL (Ruppenhofer et al., 2008; Choi et al., 2006). In general, the features aim to capture (a) local properties of the candidate opinion expressions and arguments and (b) syntactic and semantic attributes of their relation.

Words and POS tags: the words contained in the candidate and their POS tags.
Lexicon: For each word in the candidate, we include its WordNet hypernyms

[1]We randomly split the training data into 10 parts and obtained the 50-best CRF predictions on each part for the generation of candidates. We also experimented with candidates generated from more CRF predictions, but did not find any performance improvement for the task.
[2]We use the Stanford Parser to generate parse trees and dependency graphs.

and its strength of subjectivity in the Subjectivity Lexicon[3] (e.g., weaksubj, strongsubj).

Phrase type: the syntactic category of the deepest constituent that covers the candidate in the parse tree, e.g., NP, VP.

Semantic frames: For each verb in the opinion candidate, we include its frame types according to FrameNet[4].

Distance: the relative distance (number of words) between the opinion and argument candidates.

Dependency Path: the shortest path in the dependency tree between the opinion candidate and the target candidate, e.g., ccomp↑nsubj↑. We also include word types and POS types in the paths, e.g., *opinion*$\uparrow_{ccomp}$*suffering*$\uparrow_{nsubj}$*patient*, NN$\uparrow_{ccomp}$VBG$\uparrow_{nsubj}$NN. The dependency path has been shown to be very useful in extracting opinion expressions and opinion holders (Johansson and Moschitti, 2010a).

Opinion-Implicit-Arg Relations

When the *opinion-arg* relation classifier predicts that there is no suitable argument for the opinion expression candidate, it does not capture the possibility that an opinion candidate may associate with an implicit argument. To incorporate knowledge of implicit relations, we build an *opinion-implicit-arg* classifier to identify an opinion candidate with an implicit argument based on its own properties and context information.

For training, we consider all gold-standard opinion expressions as training

[3] http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

[4] https://framenet.icsi.berkeley.edu/fndrupal/

examples — including those with implicit arguments — as positive examples and those associated with explicit arguments as negative examples. For features, we use words, POS tags, phrase types, lexicon and semantic frames (see Section 3.2.2 for details) to capture the properties of the opinion expression, and also features that capture the context of the opinion expression:

Neighboring constituents: The words and grammatical roles of neighboring constituents of the opinion expression in the parse tree — the left and right sibling of the deepest constituent containing the opinion expression in the parse tree.

Parent Constituent: The grammatical role of the parent constituent of the deepest constituent containing the opinion expression.

Dependency Argument: The word types and POS types of the arguments of the dependency patterns in which the opinion expression is involved. We consider the same dependency patterns that are used to generate candidates for *opinion-arg* classification.

3.2.3 Integer Linear Programming Formulation

The goal of joint inference is to find the optimal prediction for both opinion entity extraction and opinion relation extraction. For a given sentence, we denote $\mathcal{P}$ as a set of opinion candidates, $\mathcal{A}_k$ as a set of argument candidates, where k denotes the type of opinion relation — IS-ABOUT or IS-FROM — and $\mathcal{S}$ as a set of within-sentence spans that cover all of the opinion candidates and argument

candidates. We introduce binary variable x_{iz}, where $x_{iz} = 1$ means span i is associated with label z. We also introduce binary variable u_{ij} for every pair of opinion candidate i and argument candidate j, where $u_{ij} = 1$ means i forms an opinion relation with j, and binary variable v_{ik} for every opinion candidate i in relation type k, where $v_{ik} = 1$ means i associates with an implicit argument in relation k. Given the binary variables x_{iy}, u_{ij}, v_{ik}, it is easy to recover the entity and relation assignment by checking which spans are labeled as opinion entities, and which opinion span and argument span form an opinion relation.

The objective function is defined as a linear combination of the potentials from different predictors with a parameter λ to balance the contribution of two components: opinion entity extraction and opinion relation extraction.

$$\arg\max_{x,u,v} \lambda \sum_{i \in \mathcal{S}} \sum_{z} f_{iz} x_{iz} + (1 - \lambda) \sum_{k} \sum_{i \in \mathcal{P}} \left(\sum_{j \in \mathcal{A}_k} r_{ij} u_{ij} + r_{i\emptyset} v_{ik} \right) \tag{3.1}$$

It is subject to the following linear constraints:

Constraint 1: *Uniqueness*. For each span i, we must assign one and only one label z, where $z \in \{H, D, T, O\}$.

$$\sum_{z} x_{iz} = 1$$

Constraint 2: *Non-overlapping*. If two spans i and j overlap, then at most one of the spans can be assigned to a non-NONE entity label: H, D, T.

$$\sum_{z \neq O} x_{iz} + \sum_{z \neq O} x_{jz} \leq 1$$

Constraint 3: *Consistency between the opinion-arg and opinion-implicit-arg classifiers*. For an opinion candidate i, if it is predicted to have an implicit argument in relation k, $v_{ik} = 1$, then no argument candidate should form a relation with i. If $v_{ik} = 0$, then there exists some argument candidate $j \in \mathcal{A}_k$ such that $u_{ij} = 1$. We introduce two auxiliary binary variables a_{ik} and b_{ik} to limit the maximum

number of relations associated with each opinion candidate to be less than or equal to three[5]. When $v_{ik} = 1$, a_{ik} and b_{ik} have to be 0.

$$\sum_{j \in \mathcal{A}_k} u_{ij} = 1 - v_{ik} + a_{ik} + b_{ik}$$
$$a_{ik} \leq 1 - v_{ik}, b_{ik} \leq 1 - v_{ik}$$

Constraint 4: *Consistency between opinion-arg classifier and opinion entity extractor*. Suppose an argument candidate j in relation k is assigned an argument label by the entity extractor, that is $x_{jz} = 1$ ($z = T$ for IS-ABOUT relation and $z = H$ for IS-FROM relation), then there exists some opinion candidates that associate with j. Similar to constraint 3, we introduce auxiliary binary variables c_j and d_j to enforce that an argument j links to at most three opinion expressions. If $x_{jz} = 0$, then no relations should be extracted for j.

$$\sum_{i \in \mathcal{P}} u_{ij} = x_{jz} + c_{jk} + d_{jk}$$
$$c_{jk} \leq x_{jz}, d_{jk} \leq x_{jz}$$

Constraint 5: *Consistency between the opinion-implicit-arg classifier and opinion entity extractor*. When an opinion candidate i is predicted to associate with an implicit argument in relation k, that is $v_{ik} = 1$, then we allow x_{iD} to be either 1 or 0 depending on the confidence of labeling i as an opinion expression. When $v_{ik} = 0$, there exisits some opinion argument associated with the opinion candidate, and we enforce $x_{iD} = 1$, which means the entity extractor agrees to label i as an opinion expression.

$$v_{ik} + x_{iD} \geq 1$$

Note that in our ILP formulation, the label assignment for a candidate span involves one multiple-choice decision among different opinion entity labels and

[5]It is possible to add more auxiliary variables to allow more than three arguments to link to an opinion expression, but this rarely happens in our experiments. For the IS-FROM relation, we set $a_{ik} = 0, b_{ik} = 0$ since an opinion expression usually has only one holder.

the "NONE" entity label. The scores of different label assignments are comparable for the same span since they come from one entity extraction model. This makes our ILP formulation advantageous over the ILP formulation proposed in Choi et al. (2006), which needs m binary decisions for a candidate span, where m is the number of types of opinion entities, and the score for each possible label assignment is obtained by the sum of raw scores from m independent extraction models. This design choice also allows us to easily deal with multiple types of opinion arguments and opinion relations.

3.3 Experiments

3.3.1 Experimental Setup

For evaluation, we used the NRRC Multi-Perspective Question Answering (MPQA) corpus (Wiebe et al., 2005; Wilson, 2008a), a widely used data set for fine-grained opinion analysis.[6] We considered the subset of 482 documents[7] that contain attitude and target annotations. There are a total of 9,471 sentences with opinion-related labels at the phrase level. We set aside 132 documents as a development set and use 350 documents as the evaluation set. All experiments employ 10-fold cross validation on the evaluation set; the average over the 10 runs is reported.

Our gold standard opinion expressions, opinion targets and opinion holders correspond to the direct subjective annotations, target annotations, and agent

[6] Available at http://www.cs.pitt.edu/mpqa/.

[7] 349 news articles from the original MPQA corpus, 84 Wall Street Journal articles (Xbank), and 48 articles from the American National Corpus.

	Opinion	Target	Holder
TotalNum	5849	4676	4244
	Opinion-arg Relations		Implicit Relations
IS-ABOUT	4823		1302
IS-FROM	4662		1187

Table 3.1: Statistics of the MPQA Corpus.

annotations, respectively. The IS-FROM relation is obtained from the *agent* attribute of each opinion expression. The IS-ABOUT relation is obtained from the attitude annotations: each opinion expression is annotated with attitude frames, and each attitude frame is associated with a list of targets. The relations may overlap: for example, in the following sentence, the target of relation 1 contains relation 2.

> [John]$_{H_1}$ is happy$_{O_1}$ because [[he]$_{H_2}$ loves$_{O_2}$ [being at Enderly Park]$_{T_2}$]$_{T_1}$.

We discard relations that contain sub-relations because we believe that identifying the sub-relations usually is sufficient to recover the discarded relations. (Prediction of overlapping relations is considered as future work.) In the example above, we will identify (*loves*, *being at Enderly Park*) as an IS-ABOUT relation and *happy* as an opinion expression associated with an implicit target. Table 3.1 shows some statistics of the corpus.

We use precision, recall, and F-measure to evaluate the quality of the model. Precision is defined as $\frac{|C \cap P|}{|P|}$ and recall, as $\frac{|C \cap P|}{|C|}$, where C and P are the sets of correct and predicted text spans, respectively. F-measure is computed as $\frac{2PR}{P+R}$. Because the boundaries of opinion expressions are hard to define even for human annotators (Wiebe et al., 2005), previous research mainly focused on soft

precision and recall measures for performance evaluation. Breck et al. (2007b) used a metric that considers a predicted entity span to be correct if it overlaps with a correct entity span. We refer to it as the *overlap* metric. And we refer to the metric that considers a predicted entity span to be correct if it exactly matches a correct entity span as the *exact* metric.

3.3.2 Implementation Details

We trained CRFs for opinion entity extraction using the following features: indicators for words, POS tags, and lexicon features (the subjectivity strength of the word in the Subjectivity Lexicon). All features are computed for the current token and tokens in a $[-1, +1]$ window. We used L2-regularization; the regularization parameter was tuned using the development set. We trained the classifiers for relation extraction using L1-regularized logistic regression with default parameters using the LIBLINEAR (Fan et al., 2008) package. For joint inference, we used GLPK[8] to provide the optimal ILP solution. The parameter λ was tuned using the development set.

3.3.3 Baselines

We compare our approach to several pipeline baselines. Each extracts opinion entities first using the same CRF employed in our approach, and then predicts opinion relations on the opinion entity candidates obtained from the CRF prediction. Three relation extraction techniques were used in the baselines:

[8] `http://www.gnu.org/software/glpk/`

- Adj: Inspired by the adjacency rule used in Hu and Liu (2004a), it links each argument candidate to its nearest opinion candidate. Arguments that do not link to any opinion candidate are discarded. This is also used as a strong baseline in Choi et al. (2006).
- Syn: Links pairs of opinion and argument candidates that present prominent syntactic patterns. (We consider the syntactic patterns listed in Section 3.2.2.) Previous work also demonstrates the effectiveness of syntactic information in opinion extraction (Johansson and Moschitti, 2013a).
- RE: Predicts opinion relations by employing the *opinion-arg* classifier and *opinion-implicit-arg* classifier. First, the *opinion-arg* classifier identifies pairs of opinion and argument candidates that form valid opinion relations, and then the *opinion-implicit-arg* classifier is used on the remaining opinion candidates to further identify opinion expressions without explicit arguments.

We report results using opinion entity candidates from the best CRF output and the merged 10-best CRF output.[9] The motivation of merging the 10-best output is to increase recall for the pipeline methods.

3.3.4 Results

Table 3.2 shows the results of opinion entity extraction using both *overlap* and *exact* metrics. We compare our approach with the pipeline baselines and CRF

[9]It is similar to the *merged 10-best* baseline in Choi et al. (2006). If an entity E_i extracted by the ith-best sequence overlaps with an entity E_j extracted by the jth-best sequence, where $i \leq j$, then we discard E_j. If E_i and E_j do not overlap, then we consider both entities.

	Opinion Expression			Opinion Target			Opinion Holder		
Method	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
CRF	82.21	66.15	73.31	73.22	48.58	58.41	72.32	49.09	58.48
CRF+Adj	82.21	66.15	73.31	80.87	42.31	55.56	75.24	48.48	58.97
CRF+Syn	82.21	66.15	73.31	81.87	30.36	44.29	78.97	40.20	53.28
CRF+RE	83.02	48.99	61.62	85.07	22.01	34.97	78.13	40.40	53.26
Joint-Model	71.16	77.85	**74.35***	75.18	57.12	**64.92****	67.01	66.46	**66.73****
CRF	66.60	52.57	58.76	44.44	29.60	35.54	65.18	44.24	52.71
CRF+Adj	66.60	52.57	58.76	49.10	25.81	33.83	68.03	43.84	53.32
CRF+Syn	66.60	52.57	58.76	50.26	18.41	26.94	74.60	37.98	50.33
CRF+RE	69.27	40.09	50.79	60.45	15.37	24.51	75	38.79	51.13
Joint-Model	57.39	62.40	**59.79***	49.15	38.33	**43.07****	62.73	62.22	**62.47****

Table 3.2: Performance on opinion entity extraction using the *overlap* and *exact* matching metrics (the top table uses *overlap* and the bottom table uses *exact*). Two-tailed t-test results are shown on F1 measure for our method compared to the other baselines (statistical significance is indicated with $^{*}(p < 0.05)$, $^{**}(p < 0.005)$).

(the first step of the pipeline). We can see that our joint inference approach significantly outperforms all the baselines in F1 measure on extracting all types of opinion entities. In general, by adding the relation extraction step, the pipeline baselines can improve precision over the CRF but fail at recall. CRF+Syn and CRF+Adj provide the same performance as CRF since the relation extraction step only affects the results of opinion arguments. By incorporating syntactic information, CRF+Syn provides better precision than CRF+Adj on extracting arguments at the expense of recall. This indicates that using simple syntactic rules would mistakenly filter many correct relations. By using binary classifiers to predict relations, CRF+RE produces high precision on opinion and target extraction but also results in very low recall. Using the *exact* metric, we observe the same general trend in the results as the *overlap* metric. The scores are lower since the metric is much stricter.

Table 3.3 shows the results of opinion relation extraction using the *overlap* metric. We compare our approach with pipelined baselines in two settings: one

	IS-ABOUT			IS-FROM		
Method	**P**	**R**	**F1**	**P**	**R**	**F1**
CRF+Adj	73.65	37.34	49.55	70.22	41.58	52.23
CRF+Syn	76.21	28.28	41.25	77.48	36.63	49.74
CRF+RE	78.26	20.33	32.28	74.81	37.55	50.00
CRF+Adj-merged-10-best	25.05	61.18	35.55	30.28	62.82	40.87
CRF+Syn-merged-10-best	41.60	45.66	43.53	48.08	54.03	50.88
CRF+RE-merged-10-best	51.60	33.09	40.32	47.73	54.40	50.84
Joint-Model	64.38	51.20	**57.04****	64.97	58.61	**61.63****

Table 3.3: Performance on opinion relation extraction using the *overlap* metric.

employs relation extraction on 1-best output of CRF (top half of table) and the other employs the merged 10-best output of CRF (bottom half of table). We can see that, in general, using merged 10-best CRF outputs boosts the recall while sacrificing precision. This is expected since merging the 10-best CRF outputs favors candidates that are believed to be more accurate by the CRF predictor. If CRF makes mistakes, the mistakes will propagate to the relation extraction step. The poor performance on precision further confirms the error propagation problem in the pipeline approaches. In contrast, our joint-inference method successfully boosts the recall while maintaining reasonable precision. This demonstrates that joint inference can effectively leverage the advantage of individual predictors and limit error propagation.

To demonstrate the effectiveness of different potentials in our joint inference model, we consider three variants of our ILP formulation that omit some potentials in the joint inference: one is ILP-W/O-ENTITY, which extracts opinion relations without integrating information from opinion entity extraction; one is ILP-W-SINGLE-RE, which focuses on extracting a single opinion relation and ignores the information from the other relation; the third one is ILP-W/O-IMPLICIT-RE, which omits the potential for *opinion-implicit-arg* relation and assumes every opinion expression is linked to an explicit argument. The objective function

of ILP-W/O-ENTITY can be represented as

$$\arg\max_{u} \sum_{k} \sum_{i \in O} \sum_{j \in \mathcal{A}_k} r_{ij} u_{ij} \tag{3.2}$$

which is subject to constraints on u_{ij} to enforce relations to not overlap and limit the maximum number of relations that can be extracted for each opinion expression and each argument. For ILP-W-SINGLE-RE, we simply remove the variables associated with one opinion relation in the objective function (3.1) and constraints. The formulation of ILP-W/O-IMPLICIT-RE removes the variables associated with potential r_i in the objective function and corresponding constraints. It can be viewed as an extension to the ILP approach in Choi et al. (2006) that includes opinion targets and uses simpler ILP formulation with only one parameter and fewer binary variables and constraints to represent entity label assignments [10].

	IS-ABOUT **Relation Extraction**			IS-FROM **Relation Extraction**		
Method	**P**	**R**	**F1**	**P**	**R**	**F1**
ILP-W/O-ENTITY	49.10	40.48	44.38	44.77	58.24	50.63
ILP-W-SINGLE-RE	63.88	49.35	55.68	53.64	65.02	58.78
ILP-W/O-IMPLICIT-RE	62.00	44.73	51.97	73.23	51.28	60.32
Joint-Model	64.38	51.20	**57.04****	64.97	58.61	**61.63***

Table 3.4: Performance comparison of our joint approach with other joint approaches.

Table 3.4 shows the results of these methods on opinion relation extraction. We can see that without the knowledge of the entity extractor, ILP-W/O-ENTITY provides poor performance on both relation extraction tasks. This confirms the effectiveness of leveraging knowledge from entity extractor and relation extractor. The improvement yielded by our approach over ILP-W-SINGLE-RE demonstrates the benefit of jointly optimizing different types of opinion relations. Our

[10] We compared the proposed ILP formulation with the ILP formulation in Choi et al. (2006) on extracting opinion holders, opinion expressions and IS-FROM relations, and showed that the proposed ILP formulation performs better on all three extraction tasks.

approach also outperforms ILP-W/O-IMPLICIT-RE, which does not take into account implicit relations. The results demonstrate that incorporating knowledge of implicit opinion relations is important.

3.3.5 Discussion

We note that the joint inference model yields a clear improvement on recall but not on precision compared to the CRF-based baselines. Analyzing the errors, we found that the joint model extracts comparable number of opinion entities compared to the gold standard, while the CRF-based baselines extract significantly fewer opinion entities (around 60% of the number of entities in the gold standard). With more extracted opinion entities, the precision is sacrificed but recall is boosted substantially, and overall we see an increase in F-measure. We also found that a good portion of errors were made because the generated candidates failed to cover the correct solutions. Recall that the joint model finds the global optimal solution over a set of opinion entity and relation candidates, which are obtained from the n-best CRF predictions and constituents in the parse tree that satisfy certain syntactic patterns. It is possible that the generated candidates do not contain the gold standard answers. For example, our model failed to identify the IS-ABOUT relation *(offers, general aid)* from the following sentence *Powell had contacted ... and received* $\underline{offers}_{O_1}$ *of* $[general\ aid]_{T_1}$*...* because both the CRF predictor and syntactic heuristics fail to capture *(offers, general aid)* as a potential relation candidate. By applying simple heuristics such as treating all verbs or verb phrases as opinion candidates would not help because it would introduce a large number of negative candidates and lower the accuracy of relation extraction (only 52% of the opinion expressions are verbs or verb phrases and 64%

of the opinion targets are noun or noun phrases in the corpus we used). Therefore, a more effective candidate generation method is needed to allow more candidates while limiting the number of negative candidates. We also found incorrect parsing to be a cause of errors. We hope to study ways to account for such errors in our approach as future work.

For computational time, our ILP formulation can be solved very efficiently using advanced ILP solvers. In our experiment, using GLPK's branch-and-cut solver took 0.2 seconds to produce optimal ILP solutions for 1000 sentences on a machine with Intel Core 2 Duo CPU and 4GB RAM.

3.4 Chapter Summary

In this chapter, we proposed a joint inference approach for opinion entity and relation extraction. It jointly optimizes opinion entity extraction and opinion relation extraction using integer linear programming with constraints that enforce global consistency. We showed that our approach could effectively integrate information from different predictors and achieve significant improvements on individual tasks.

CHAPTER 4

JOINT OPINION EXPRESSION EXTRACTION AND ATTRIBUTE CLASSIFICATION

Besides entities and relations, we also want to extract semantic attributes of the opinions. In this chapter, we propose joint models for extracting opinion expressions along with their polarity and intensity. Consider the following sentence[1] for example,

He was in favor of $_{medium}$ the rebels despite being severely criticized $_{high}$.

We want to simultaneously extract the opinion expressions "in favor of" and "being severely criticized" along with their polarity and intensity values: one has positive polarity, medium intensity, and the other has negative polarity, high intensity.

Most existing approaches tackle the tasks of opinion expression extraction and attribute classification in isolation. The first task is typically formulated as a sequence labeling problem, where the goal is to label the boundaries of text spans that correspond to opinion expressions (Breck et al., 2007a). The second task is usually treated as a binary or multi-class classification problem (Wilson et al., 2005b; Choi and Cardie, 2008; Yessenalina and Cardie, 2011), where the goal is to assign a class label to a text fragment (e.g., a phrase or a sentence). Solutions to the two tasks can be applied in a pipeline architecture to extract opinion expressions and their attributes. However, pipeline systems suffer from error propagation: opinion expression errors propagate and lead to unrecoverable errors in attribute classification.

[1]We use colored boxes to mark the textual spans of opinion expressions where green (red) denotes positive (negative) polarity and use subscripts to denote intensity.

Limited work has been done on the joint modeling of opinion expression extraction and attribute classification. Choi and Cardie (2010) first proposed a joint sequence labeling approach to extract opinion expressions and label them with polarity and intensity. Their approach treats both expression extraction and attribute classification as token-level sequence labeling tasks, and thus cannot model the label distribution over expressions even though the annotations are given at the expression level. Johansson and Moschitti (2011) considered a pipeline of opinion extraction followed by polarity classification and propose re-ranking its k-best outputs using global features. One key issue, however, is that the approach enumerates the k-best output in a pipeline manner and thus they do not necessarily correspond to the k-best global decisions. Moreover, as the number of opinion attributes grows, it is not clear how to identify the best k for each attribute.

Our contribution in this chapter is twofold. First, we propose a semi-CRF-based model for segment-level opinion expression extraction. It allows easy incorporation of phrase-level evidence into the model and allows effective segment candidate generation using parsing information. We evaluate our model on two opinion expression extraction tasks: identifying direct subjective expressions (DSEs) and expressive subjective expressions (ESEs). Experimental results show that our approach outperforms the standard CRF-based approach for the task by a large margin. This work was published in Yang and Cardie (2012).

Second, we extend the semi-CRF-based model to account for segment-level dependencies between opinion expressions and opinion attributes. Specifically, we consider two kinds of joint models: (1) *joint learning*, which estimates the segmentation- and attribute-specific parameters jointly during training; and (2)

joint inference, which estimates the segmentation- and attribute-specific parameters separately during training and combines them only during inference time. Extensive experiments on the MPQA corpus (Wiebe et al., 2005) shown that both joint models provide substantial improvements over the previously published results. Error analysis provides additional understanding of the differences between joint learning and joint inference and suggests that joint inference can be more effective and more efficient for the task in practice. This work was published in Yang and Cardie (2014b).

4.1 Related Work

In this section, we provide a detailed overview of the related work on opinion expression extraction and phrase-level polarity classification, as well as previous work on joint modeling of these two problems.

4.1.1 Opinion Expression Extraction

Earlier research to extract opinion expressions mainly focused on single-word expressions (Wiebe et al., 2005; Wilson et al., 2005a; Munson et al., 2005). More recent studies tackle the extraction of opinion phrases or longer opinion expressions. Breck et al. (2007b) formulated the problem as a token-level sequence labeling problem; their conditional random fields (CRF) based approach was shown to significantly outperform two subjectivity-clue-based baselines. Others also employed the CRF model for identifying opinion holders (Choi et al., 2005) and opinion expressions (Choi and Cardie, 2010). Johansson and Mos-

chitti (2010b; 2011) shown that encoding word dependency relations in the output of a token-level sequence labeler can further improve the extraction of opinion expressions. All of the above approaches, however, cannot easily incorporate phrase-level evidence (e.g., "as usual" is usually an opinion-bearing phrase) into the learning model.

In our work, we employ a segment-level sequence labeler based on semi-Markov CRFs that can incorporate rich phrase-level features. Semi-CRFs (Sarawagi and Cohen, 2004) are general CRFs that relax the Markovian assumptions to allow sequence labeling at the segment level. Previous work has shown that semi-CRFs are superior to CRFs for NER and Chinese word segmentation (Sarawagi and Cohen, 2004; Okanohara et al., 2006; Andrew, 2006). The task of opinion expression extraction is known to be harder than traditional NER since subjective expressions exhibit substantial lexical variation, and their recognition requires more attention to linguistic structure.

4.1.2 Phrase-level Polarity Classification

Wilson et al. (2005b) first motivated and studied phrase-level polarity classification on an open-domain corpus. Choi and Cardie (2008) developed inference rules to capture compositional effects at the lexical level on phrase-level polarity classification. Yessenalina and Cardie (2011) and Socher et al. (2013) learn continuous-valued phrase representations by combining the representations of words within an opinion expression and using them as features for classifying polarity and intensity. All of these approaches assume the opinion expressions are available before training the classifiers. However, in real-world settings, the

spans of opinion expressions within the sentence are not available. In fact, Choi and Cardie (2008) demonstrated that the performance of expression-level polarity classification degrades as more surrounding (but irrelevant) context is considered. This motivates the additional task of identifying the spans of opinion expressions.

4.1.3 Joint Modeling

There has been limited work on the joint modeling of opinion expression extraction and attribute classification. Choi and Cardie (2010) first developed a joint sequence labeler that jointly tags opinions, polarity and intensity by training CRFs with hierarchical features (Zhao et al., 2008). One major drawback of their approach is that it models both opinion extraction and attribute labeling as tasks in token-level sequence labeling, and thus cannot model their interactions at the expression-level. Johansson and Moschitti (2011) and Johansson and Moschitti (2013b) propose a joint approach to opinion expression extraction and polarity classification by re-ranking its k-best output using global features. One major issue with their approach is that the k-best candidates were obtained without global reasoning about the relative uncertainty in the individual stages. As the number of considered attributes grows, it also becomes harder to decide how many predictions to select from each attribute classifier. Compared to the existing approaches, our joint models have the advantage of modeling opinion expression extraction and attribute classification at the segment level.

4.2 A semi-CRF-based Model for Opinion Expression Extraction

In this section, we describe a semi-CRF-based model for segment-level opinion expression extraction. We will describe how to extend this model to jointly infer opinion expressions and their attributes in the next section.

Unlike previous sequence labeling approaches to opinion expression extraction (e.g., Breck et al. (2007b)), we aim to model segment-level, rather than token-level, information. In particular, we explore the use of semi-Markov Conditional Random Fields (semi-CRF), which can assign labels to segments instead of tokens; hence, features can be defined at the segment level. For example, features like $[\![X \text{ is a verb phrase}]\!]$ can be easily encoded in the model.

4.2.1 An Overview of Semi-Markov CRFs

In semi-CRFs, each observed sentence x is represented as a sequence of consecutive segments $s = \langle s_1, ..., s_n \rangle$, where s_i is a triple $s_i = (t_i, u_i, y_i)$, t_i denotes the start position of segment s_i, u_i denotes the end position, and y_i denotes the label of the segment. Segments are restricted to have positive length less than or equal to a maximum length of L that has been seen in the corpus ($1 \leq u_i - t_i + 1 \leq L$).

Features in semi-CRFs are defined at the segment level rather than the word level. The feature function $g(i, x, s)$ is a function of x, the current segment s_i, and the label y_{i-1} of the previous segment s_{i-1} (we consider the usual first-order Markovian assumption). It can also be written as $g(x, t_i, u_i, y_i, y_{i-1})$. The condi-

tional probability of a segmentation s given a sequence x is defined as

$$p(s|x) = \frac{1}{Z(x)} \exp\left\{\sum_i \sum_k \lambda_k g_k(i, x, s)\right\} \tag{4.1}$$

where

$$Z(x) = \sum_{s' \in S} \exp\left\{\sum_i \sum_k \lambda_k g_k(i, x, s')\right\}$$

and the set S contains all possible segmentations obtained from segment candidates with length ranging from 1 to the maximum length L.

The correct segmentation s of a sentence is defined as a sequence of entity segments (i.e., the entities to be extracted), and non-entity segments that are all unit-length segments.

4.2.2 Segment-level Opinion Expression Extraction

We now describe an extension of the standard semi-CRFs for segment-level opinion expression extraction. Standard semi-CRFs make the assumption that there is a fixed maximum length L for all entities. In practice, L is usually set as the length of the longest entities seen during training. This usually does not apply to the entities seen during prediction time. Especially for opinion expressions, L is unbounded because an opinion expressions may be a clause or a whole sentence, which can be arbitrarily long. Thus, fixing an upper bound on segment length based on the observed entities may lead to an incorrect removal of segments during inference. Also, note that possible segment candidates are generated based on the length constraint, which means any span of the text consisting of no more than L words would be considered as a possible segment. This would lead to the consideration of implausible segments.

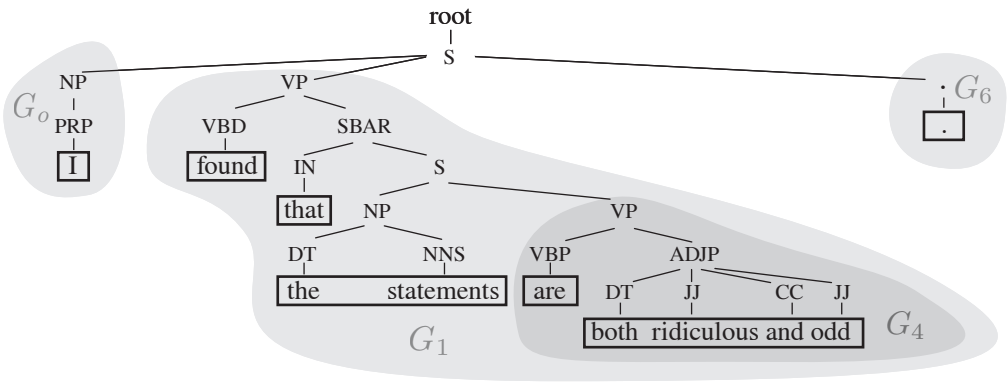

Figure 4.1: A parse tree example. There are seven segment units in the sentence. The shaded regions correspond to segment groups, where G_i represents the segment group starting from segment unit U_i.

To address these problems, we propose techniques to incorporate parsing information into the modeling of segments in semi-CRFs. More specifically, we construct segment units from the parse tree of each sentence[2], and then build up possible segment candidates based on those units. In the parse tree, each leaf phrase or leaf word is considered to be a segment unit. Each segment unit performs as the smallest unit in the model (words within a segment unit will be automatically assigned the same label). The segment units are highlighted in rectangles in the parse tree example in Figure 4.1. As the segment unitsß are not separable, we avoid implausible segments, which truncate multi-word expressions. For example, "both ridiculous and", would not be considered a possible segment in our model.

To generate segment candidates for the model, we consider meaningful combinations of consecutive segment units. Intuitively, a sentence is made up of several parts, and each has its own grammatical role or meaning. We define the boundary of these parts based on the parse tree structure. Specifically, we con-

[2]We use the Stanford Parser `http://nlp.stanford.edu/software/lex-parser.shtml` to generate the parse trees.

sider each segment unit to belong to a meaningful group defined by the span of its parent node. Two consecutive segment units are considered to belong to the same group if the subtrees rooted in their parent nodes have the same rightmost child. For example, in Figure 4.1, segment units "are" and "both ridiculous and odd" belong to the same group, while "I" and "found" belong to different groups.

Algorithm 1: Construction of segment candidates

Input: A training sentence x

Output: A set of segment candidates S

1: Obtain the segment units $U = (U_1, ..., U_m)$ by preorder traversal of the parse tree T, each U_i corresponds to a node in T
2: **for** $i = 1$ to m **do**
3: $\quad j \leftarrow i$
4: $\quad$ **while** $j \leq m$ and commonGroup$(U_i, ..., U_j)$ **do**
5: $\quad\quad j \leftarrow j + 1$
6: $\quad j \leftarrow j - 1$
7: $\quad$ **for** $k = i$ to j **do**
8: $\quad\quad s \leftarrow$ segment$(U_i, ..., U_k)$
9: $\quad\quad S \leftarrow S \cup s$
10: Return S

Following this idea, we generate possible segment candidates by Algorithm 1. Starting from each segment unit U_i, we first find the rightmost segment unit U_j that belongs to the same group as U_i. Function commonGroup$(U_i, ..., U_j)$ returns True if $U_i, ..., U_j$ are within the same group (the parent nodes of U_i,...,U_j

have the same rightmost child in their subtrees), otherwise it returns False. Then we enumerate all possible combinations of segment units $U_i, ..., U_k$ where $i \leq k \leq j$. segment$(U_i, ..., U_j)$ denotes the segment obtained by concatenating words in the consecutive segment units U_i,...,U_j. This way, segment candidates are generated without constraints on length and are meaningful for learning entity boundaries.

Based on the generated segment candidates, the correct segmentation for each training sentence can be obtained as follows. For opinion expressions that do not match any segment candidate, we break them down into smaller segments using a greedy matching process. Starting from the start position of the expression, we search for the longest candidate that is contained in the expression, add it to the correct segmentation for the sentence, set the start position to be the next position, and repeat the process. Using this process, the correct segmentation of the sentence in Figure 4.1 would be s = ⟨(I,NONE),(found,NONE),(that,NONE),(the statements,NONE),(are both rediculous and odd,OPINION),(.)⟩. Note that here non-entities correspond to segment units instead of single-word segments in the original semi-CRF model.[3]

After obtaining the set of possible segment candidates and the correct segmentation s for each training sentence, the semi-CRF model can be trained. The goal of learning is to find the optimal parameter λ by maximizing log-likelihood. We use the limited-memory BFGS algorithm (Liu and Nocedal, 1989) for optimization in our implementation, where the gradient of the log-likelihood L

[3]There are cases where words within a segment unit have different labels. This may be due to errors by the human annotators or the errors in the parser. In such cases, we consider each word within the segment unit as a segment.

(corresponding to one instance x) is computed:

$$\frac{\partial L}{\partial \lambda_k} = \sum_i g_k(x, t_i, u_i, y_i, y_{i-1}) - \sum_{s' \in S} \sum_{y,y'} \sum_j g_k(x, t'_j, u'_j, y, y') p(y, y'|x) \tag{4.2}$$

where S is all possible segmentations consisting of the generated segment candidates, $p(y, y'|x)$ is the probability of having label y for the current segment s'_j (with boundary (t'_j, u'_j)) and label y' for the previous segment s'_{j-1}.

We use a forward-backward algorithm to compute the marginal distribution $p(y, y'|x)$ and the normalization factor $Z(x)$ efficiently. For inference we seek the best segmentation $s^* = \arg\max_s p(s|x)$, where $p(s|x)$ is defined by Equation 4.1. We implement efficient inference using an extension of Viterbi algorithm to segments. In particular, define $V(j, y)$ as the largest unnormalized probability of $p(s_{1:j}|x)$ with label y at the ending position j. Then we have

$$V(j, y) = \max_{(i,j) \in s_{:,j}} \max_{y'} \phi(x, i, j, y, y') V(i-1, y')$$

where

$$\phi(x, i, j, y, y') = \exp\left\{\sum_k \lambda_k g_k(x, i, j, y, y')\right\}$$

and $s_{:,j}$ denotes the set of the generated segment candidates ending at position j. The best segmentation can be obtained from tracing the path of $\max_y V(n, y)$.

4.2.3 Features

For the features, we include CRF-style features that are segment-level extensions of the token-level features. We also include new segment-level features that can be naturally represented in semi-CRFs but not CRFs.

For CRF-style features, we consider the string representation of the current word, its part-of-speech, and a dictionary-derived feature, which is based on a

subjectivity lexicon provided by Wilson et al. (2005c). The lexicon consists of a set of words that can act as strong or weak cues to subjectivity. If the current word appears as an entry in the lexicon, then a feature *strong* or *weak* will be fired if the entry is of that strength. These features have been successfully employed in previous work (Breck et al., 2007b). To employ them in our model, we simply extend the feature definition to the segment level. For example, a token-level feature $[\![x \text{ is } great \,]\!]$ will be extended to a segment-level feature $[\![s \text{ contains } great \,]\!]$.

Previous work on semi-CRFs has explored features such as the length of the segment, the position of the segment in the current segmentation (at the beginning or the end), indicators for the start word and end word within the segment, and indicators for words before and after the segment. These features have been shown useful for the task of NE recognition (Sarawagi and Cohen, 2004; Okanohara et al., 2006). However, we only found the position of the segment to be helpful for the extraction of opinion expressions, probably due to the lack of patterns in the length distribution and word choices of opinion expressions.

Besides the above features, we design new segment-level syntactic features to capture the syntactic patterns of opinion expressions. Syntactic patterns are often used to identify useful information in information extraction tasks. In our task, we found that the majority of opinion expressions involve verb phrases.[4] For example, "was encouraged", "expressed goodwill", "cannot accept" are all within a VP constituent. To capture such structural preferences, we define several syntax-based parse features for VP-related constituents.[5]

[4]The percentages of opinion expressions involving VP/NP/PP are 53.7%/21.7%/8.8% in the data set we used.

[5]We also conducted experiments with NP and PP-related features, and could not find any performance improvement for the tasks.

Let VPROOT denote a VP constituent whose parent node is not VP, and let VPLEAF denote a VP constituent whose children nodes are non-VP. Denote the head of VPLEAF as the predicate, and its next segment unit as the argument. If a segment consists of words in the VP nodes visited by the preorder traversal from a VPROOT to a VPLEAF, then we refer to it as a verb-cluster segment. If a segment consists of a verb cluster and the argument in VPLEAF, we consider it as a VP segment. The following features are defined for verb-cluster segments and VP segments.

VPcluster: Indicates whether or not the segment matches the verb-cluster structure.

VPpred: A feature of the syntactic category and the word of the head of VPLEAF. The head of VPLEAF is the predicate of the verb phrase, which may encode some intention of opinions in the verb phrase. For example, if "warned" is the head of VPLEAF rather than "informed", the chance of the segment being an opinion expression increases.

VParg: A feature of the syntactic category and the head word of the argument in VPLEAF. For example, the noun phrase "a negative stand" is the argument of the predicate "take" in the verb phrase "take a negative stand". The argument in the verb phrase (could be a noun phrase, adjectival phrase or prepositional phrase) may convey some relevant information for identifying opinion expressions.

VPsubj: Whether the verb clusters or the argument in the segment contains an entry from the subjectivity lexicon. For example, the word "negative" is in the lexicon, so the segment "take a negative stand" has a feature ISVPSUBJ.

4.2.4 Experiments

For evaluation, we use the opinion expressions annotated in the MPQA corpus (Wiebe et al., 2005)[6]. There are two types of opinion expressions: direct subjective expressions (DSEs) — explicit mentions of private states or speech events expressing private states; and *expressive subjective expressions* (ESEs) — expressions that indicate sentiment, emotion, etc. without explicitly conveying them. Following is an example sentence labeled with DSEs and ESEs.

> The International Committee of the Red Cross, [as usual]$_{[ESE]}$, [has refused to make any statements]$_{[DSE]}$.

Table 1 shows the statistics of opinion expressions in the corpus.

	DSEs	ESEs
Sentences with opinions(%)	55.89	57.93
TotalNum	9746	11730
MaxLength	15	40
Length $\geq$ 1 (%)	43.38	71.65
Length $\geq$ 4 (%)	9.44	35.01

Table 4.1: Statistics of opinion expressions in the MPQA Corpus.

We set aside 135 documents as a development set and use 400 documents as the evaluation set. All experiments employ 10-fold cross validation on the evaluation set, and the average over all runs is reported.

We use precision, recall, and F-measure to evaluate the quality of the model. We consider the overlap matching metric (referred to as *binary matching*) defined in Section 3.3.1 and a stricter metric that computes the proportion of over-

[6]Available at http://www.cs.pitt.edu/mpqa/.

lapping spans: if a predicted expression overlaps with a correct expression, it receives a score $\frac{|s \cap s'|}{|s'|}$. We refer to this metric as *proportional matching*.

Baselines. We use the token-level CRF-based approach of Breck et al. (2007b) applied to the MPQA dataset. We employ a very similar, but not identical set of features: indicators for specific words at the current location and neighboring words in a $[-4, +4]$ window, part-of-speech features, and opinion lexicon features for tokens that are contained in the subjectivity lexicon (see Section 4.2.3). We do not include WordNet, Levin's verb categorization, and FrameNet features.

We also include two variants of standard CRFs as baselines: *segment-CRF* and *syntactic-CRF*. They incorporate segmentation information into standard CRFs without modifying the Markovian assumption. Segment-CRF treats segment units obtained from the parser as word tokens. For example, in Figure 4.1, the segment units *the statement* and *both ridiculous and odd* will be treated as word tokens. Syntactic-CRF encodes segment-level syntactic information in a standard token-level CRF as input features. We consider the VP-related segment features introduced in Section 4.2.3. VPPRE and VPARG are added to the head word of the corresponding verb phrase, and VPSUBJ and VPCLUSTER are added to each token within the corresponding segment.

Another baseline method is the original semi-CRF model (Sarawagi and Cohen, 2004). To the best of our knowledge, our work is the first to explore the use of semi-CRFs on the extraction of opinion expressions. They are considered to be more powerful than CRFs since they allow information to be represented at the expression level. The model requires an input of the maximum entity length. We set it to 15 for DSE and 40 for ESE. For segment features, we used

the same features as in our approach (see Section 4.2.3).

	DSE Extraction			ESE Extraction		
Method	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
CRF	82.83	49.38	61.87	78.56	43.57	56.05
segment-CRF	82.52	51.48	63.41	78.90	44.46	56.88
syntactic-CRF	82.48	49.09	61.55	78.41	43.39	55.95
semi-CRF	66.67	74.13	70.20	71.21	57.41	63.57
new-semi-CRF	67.72**	74.33	70.87*	73.57***	57.63	64.74**
semi-CRF(w/ syn)	64.86	74.10	69.17	70.68	56.61	62.87
new-semi-CRF(w/ syn)	70.12***	74.74*	**72.36*****	73.61***	59.27***	**65.67*****

Table 4.2: Performance on DSE and ESE extraction using binary matching. (w/ syn) indicates the inclusion of syntactic parse features VPpre, VParg and VPsubj. Results of new-semi-CRF that are statistically significantly greater than semi-CRF according to a two-tailed t-test are indicated with $^{*}(p < 0.1)$, $^{**}(p < 0.05)$, $^{***}(p < 0.005)$. T-test results are also shown for new-semi-CRF(w/ syn) versus semi-CRF(w/ syn).

	DSE Extraction			ESE Extraction		
Method	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
CRF	77.91	46.45	58.20	67.72	37.55	48.31
segment-CRF	77.86	48.58	59.83	68.03	38.34	49.04
syntactic-CRF	77.73	46.27	58.01	67.80	37.60	48.37
semi-CRF	60.38	68.34	64.11	57.30	46.20	51.16
new-semi-CRF	62.50**	68.59*	65.41*	61.69***	47.44**	53.63***
semi-CRF(w/ syn)	58.69	67.80	62.92	57.09	45.63	50.72
new-semi-CRF(w/ syn)	65.52***	68.91***	**67.17*****	61.66***	48.77***	**54.47*****

Table 4.3: Performance on DSE and ESE extraction using proportional matching. Notation is the same as above.

Results. Table 4.2 and Table 4.3 show the results of DSE and ESE extraction using two different metrics. The standard token-based CRF baseline of Breck et al. (2007b) is labeled **CRF**; the original semi-CRF baseline is labeled **semi-CRF**; and our extended semi-CRF approach is labeled **new-semi-CRF**. For semi-CRF and new-semi-CRF, the results were obtained using two different settings of features: the basic feature set includes features described in Section 4.2.3 excluding the segment-level syntactic features. In the second feature setting (labeled as **w/ syn** in the tables), we further augment the basic features with the syntactic

parse features.

Using the basic features, we observe that semi-CRF-based approaches significantly outperform CRF and its two variants segment-CRF and syntactic-CRF in F-Measure on both DSE and ESE extraction, and new-semi-CRF achieves the best results. By simply incorporating the segmentation prior into the standard CRF, segment-CRF achieves a slight improvement over standard CRF, but the results are still worse than those of semi-CRF and new-semi-CRF. However, adding segment-level syntactic features into standard CRF yields slightly reduced performance. This is not surprising as encoding segment-level information into the token-level CRF is not natural. These experiments indicate that simply encoding segmentation information into standard CRF cannot result in large performance gains. The promising F-measure results obtained by semi-CRF and new-semi-CRF confirm that relaxing the Markovian assumption on segments leads to better modeling of opinion expressions. We can also see that new-semi-CRF consistently outperforms the original semi-CRF model. This further confirms the benefit of taking into account syntactic parsing information in modeling segments. In Table 4.3, we observe the same general results trend as in Table 4.2. The scores are generally lower since proportional matching is stricter than binary matching.

We also study the impact of syntactic parse features on the semi-Markov CRF models. Here we consider the combination of VPPRE, VPARG and VPSUBJ since they turned out to be the most helpful features for our tasks. Interestingly, we found that after incorporating the syntactic parse features, performance decreases on semi-CRF. This indicates that syntactic information does not help if learning and inference take place on segment candidates generated without ac-

counting for parse information. In contrast, our approach incorporates syntactic parsing information in modeling segments and meaningful segmentations. We can see in Tables 4.2 and 4.3 that adding syntactic features successfully boosts the performance of our approach.

	DSE Extraction			ESE Extraction		
Feature set	**Precision**	**Recall**	**F-measure**	**Precision**	**Recall**	**F-measure**
Basic	67.72	74.33	70.87	73.57	57.63	64.74
Basic+VPpre	70.88	71.44	71.16	73.20	58.20	64.85
Basic+VParg	70.12	74.03	72.02	73.05	58.20	64.79
Basic+VPcluster	70.08	72.94	71.48	73.06	58.45	64.94
Basic+VPsubj	70.04	72.34	71.17	73.31	58.53	65.09
Basic+VPpre+VPsubj	70.91	72.54	71.72	73.61	58.29	65.07
Basic+VParg+VPsubj	70.45	73.53	71.96	74.45	57.80	65.07
Basic+VPpre+VParg+VPsubj	70.12	74.74	**72.36**	73.61	59.27	**65.67**
Basic+VPcluster+VPpre+VParg+VPsubj	70.91	72.54	71.72	72.84	58.45	64.86

Table 4.4: Effect of syntactic features on DSE and ESE extraction using binary matching.

To further explore the effect of the syntactic features, we include the results of our model with different configurations of syntactic features in Table 4.4 (here we focus on the binary matching metric as the results with the proportional matching metric demonstrate a similar conclusion). We can see that using the basic features and the combination of VPPRE, VPARG and VPSUBJ yields the best results for both DSE and ESE extraction. For DSE extraction, combining these three features improves the precision noticeably from 67.72% to 70.12% while the recall slightly improves. This indicates that VP-related structural information is very helpful for modeling segments as DSEs. However, this trend is not so clear for ESE extraction. This may be due to the fact that DSEs often involve verb phrases while ESEs are represented via a variety of syntactic structures.

Comparison with previous work. In Table 4.5, we compare our results to the previous work on opinion expression extraction (here we also focus on the bi-

Task	Method	F-measure
DSE Extraction	Breck et al. Baseline	70.65
	CRF+Reranking Baseline	63.87
	Our approach	72.36
	Our approach+Reranking	**73.12**
ESE Extraction	Breck et al. Baseline	63.43
	CRF+Reranking Baseline	58.21
	Our approach	65.67
	Our approach+Reranking	**67.01**

Table 4.5: Performance comparison of our work with previous work on DSE and ESE extraction using binary matching.

nary matching metric due to the similar trend demonstrated by the proportional matching metric). Breck et al. (2007b) presents the state-of-the-art sequence labeling approach on the tasks of DSE and ESE extraction. Their best results are shown as **Breck et al. Baseline** in the table. Johansson and Moschitti (2010b) used a reranking technique on the best k outputs of a sequence labeler to further improve their sequence labeling results on the task of extracting DSEs, ESEs and OSEs (Objective Speech Events) (we don't consider OSEs here). Results using our re-implementation of their approach using SVM^{struct} (Tsochantaridis et al., 2004) on the output of CRF are labeled **CRF+Reranking Baseline** in the table. We use the same features and parameter settings as in their approach. **Our approach+Reranking** are results obtained by applying the reranking step on the output of our new-semi-CRF approach.

We can see that our approach outperforms the Breck et al. Baseline on both DSE extraction and ESE extraction in spite of the fact that we do not use their WordNet, Levin's verb categorization, and FrameNet features. The CRF+Reranking Baseline does provide a performance increase over the baseline CRF results, but overall it cannot beat the other methods since the CRF baseline is very low. As one might expect, reranking also succeeds in boosting the performance of new-semi-CRF, achieving the best performance on F-measure for

both DSE and ESE extraction. Note that the inter-annotator agreement results for these two tasks are 75% for DSE and 72% for ESE using a similar metric to binary matching. Our results are much closer to these inter-annotator scores than previous systems especially for DSEs.

4.2.5 Discussion

Note that our new-semi-CRF approach outperforms the original semi-CRF w.r.t. both precision and recall, but compared to CRF, our approach yields a clear improvement on recall but not on precision. An error analysis helps explain why. We found that our semi-CRF approach predicted almost the same number of DSEs as the gold standard labels while CRF only predicted half of them (for ESE extraction, the trend is similar). With more predicted entities, the precision is sacrificed but recall is boosted substantially, and overall we see an increase in F-measure.

Looking further into the errors, we found several mistakes that could potentially be fixed to yield better a precision score. Some errors were due to the false prediction of speech events like "said" or "told" as DSEs in cases where they just introduced statements of fact without expressing any private state. Adding features to distinguish such cases should help improve performance. Other errors were due to inadequate modeling of the context surrounding the expressions. For example, "enjoy a relative advantage" was falsely predicted as an ESE. If incorporating information about the subject of this verb phrase which is "products", this mistake could be avoided since "products" cannot hold or express private state. We also noticed some errors caused by inaccurate parsing and

hope to study ways to account for these in our approach as future work.

By comparing the extraction results across different methods, we see that full parsing provides many benefits for modeling segment boundaries and improving the prediction precision for opinion expression extraction. For example, given the sentence, "... who are living [a lot better]$_{[ESE]}$ ...", both CRF and the original semi-CRF extract "lot better" as an ESE, while our approach correctly extracts "a lot better" as an ESE. And we also found many cases where the original semi-CRF cannot extract the opinion expressions while our approach can. Another benefit of utilizing parsing is to speed up learning and inference. Although in theory, the computational cost of parsing is $O(g \times n^3)$ where g is the grammar size and n is the sentence length while the cost of semi-CRFs is $O(K^2 \times L \times n)$ where K is the number of labels and L is the maximum entity length, feature extraction overhead and the potentially large number of learning iterations in parameter optimization may lead to a long training time for semi-CRFs. In our experiments on the MPQA data set, our machine with Intel Core 2 Duo CPU and 4GB RAM took 2 hours to fully parse 11,114 sentences using the Stanford Parser, and also 2 hours to train the standard semi-CRF. With the parsing information, our semi-CRF-based approach is able to finish training in 15 minutes. As full parsing would be expensive when the average sentence length is very large, it would be interesting to study how to utilize parsing with less cost in our task.

4.3 Joint Segmentation and Classification

Based on the segment-level sequence labeling approach to opinion expression extraction, we propose models that can perform joint extraction of opinion expressions and opinion attributes. Note that we do not distinguish the opinion expression type (i.e., DSE or ESE) in this work, but it can be easily incorporated as an additional attribute in our models. We consider two types of opinion attributes: polarity, which takes values from {*positive, negative, neutral*}, and intensity, which takes values from {*high, medium, low*}.

The key idea of joint modeling is to model the dependency between opinion segmentation and attribute classification. In the following, we describe how we model opinion segmentation and attribute classification separately, and then present two types of joint models.

4.3.1 Opinion Segmentation using Loss-aware Semi-CRFs

Given a sentence $\mathbf{x}$, denote an opinion segmentation as $\mathbf{y_s} = \langle (s_0, b_0), ..., (s_k, b_k) \rangle$, where the $s_{0:k}$ are consecutive segments that form a segmentation of $\mathbf{x}$; each segment $s_i = (t_i, u_i)$ consists of the positions of the start token t_i and an end token u_i; and each s_i is associated with a binary variable $b_i \in \{I, O\}$, which indicates whether it is an opinion expression (I) or not (O). Take the sentence in Figure 5.5, for example. The corresponding opinion segmentation is $\mathbf{y_s} = \langle ((0,0), O), ((1,1), O), ((2,6), I), ((7,8), O), ((9,9), O), ((10,12), I), ((13,13), O) \rangle$, where each segment corresponds to an opinion expression or to a phrase unit that does not express any opinion.

Using the semi-Markov CRF described in Section 4.2, we can define the following conditional distribution:

$$P(\mathbf{y_s}|\mathbf{x}) = \frac{\exp\{\sum_{i=1}^{|\mathbf{y_s}|} \theta \cdot f(y_{s_i}, y_{s_{i-1}}, \mathbf{x})\}}{\sum_{\mathbf{y'_s} \in \mathcal{Y}} \exp\{\sum_{i=1}^{|\mathbf{y'_s}|} \theta \cdot f(y'_{s_i}, y'_{s_{i-1}}, \mathbf{x})\}} \tag{4.3}$$

where θ denotes the model parameters, $y_{s_i} = (s_i, b_i)$ and f denotes a feature function that encodes the potentials of the boundaries for opinion segments and the potentials of transitions between two consecutive labeled segments. Note that the probability is normalized over segment candidates that are plausible according to the parsing structure of the sentence. Figure 4.2 shows some candidate segmentations generated for an example sentence. Such a technique results in a large reduction in training time and was shown to be effective for identifying opinion expressions.

The standard training objective of a semi-CRF, is to minimize the log loss

$$L(\theta) = \arg\min_{\theta} - \sum_{i=1}^{N} \log P(\mathbf{y_s^{(i)}}|\mathbf{x}^{(i)}) \tag{4.4}$$

It penalizes any predicted opinion expression whose boundaries do not exactly align with the boundaries of the correct opinion expressions using 0-1 loss. Unfortunately, exact boundary matching is often not used as an evaluation metric for opinion expression extraction since it is hard for human annotators to agree on the exact boundaries of opinion expressions.[7] Most previous work used *proportional matching* (Johansson and Moschitti, 2013b) as it takes into account the overlapping proportion of the predicted and the correct opinion expressions to compute precision and recall. To incorporate this evaluation metric into training, we use softmax-margin (Gimpel and Smith, 2010) that replace $P(\mathbf{y_s^{(i)}}|\mathbf{x}^{(i)})$ in

[7]The inter-annotator agreement on boundaries of opinion expressions is not stressed in MPQA (Wiebe et al., 2005).

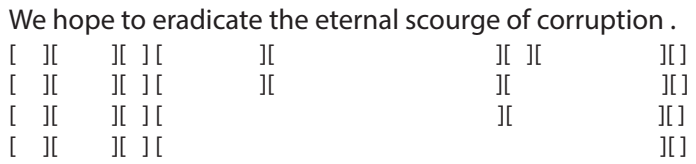

Figure 4.2: Examples of segmentation candidates

(4.4) with $P_{cost}(\mathbf{y_s^{(i)}}|\mathbf{x}^{(i)})$, which equals

$$\frac{\exp\{\sum_{i=1}^{|\mathbf{y_s}|}\theta\cdot f(y_{s_i},y_{s_{i-1}},\mathbf{x})\}}{\sum_{\mathbf{y'_s}\in\mathcal{Y}}\exp\{\sum_{i=1}^{|\mathbf{y'_s}|}\theta\cdot f(y'_{s_i},y'_{s_{i-1}},\mathbf{x})+l(\mathbf{y'_s},\mathbf{y_s})\}}$$

and we define the loss function $l(\mathbf{y'_s},\mathbf{y_s})$ as

$$\sum_{i=1}^{|\mathbf{y'_s}|}\sum_{j=1}^{|\mathbf{y_s}|}(\mathbb{1}\{b'_i\neq b_j\wedge b'_i\neq O\}\frac{|s_j\cap s'_i|}{|s'_i|}+\mathbb{1}\{b'_i\neq b_j\wedge b_j\neq O\}\frac{|s_j\cap s'_i|}{|s_j|})$$

which is the sum of the precision and recall errors of segment labeling using proportional matching. The loss-augmented probability is only computed during training. The more the proposed labeled segmentation overlaps with the true labeled segmentation for $\mathbf{x}$, the less it will be penalized.

During inference, we can obtain the best labeled segmentation by solving

$$\underset{\mathbf{y_s}}{\text{argmax}}P(\mathbf{y_s}|\mathbf{x})=\underset{\mathbf{y_s}}{\text{argmax}}\sum_{i=1}^{|\mathbf{y_s}|}\theta\cdot f(y_{s_i},y_{s_{i-1}},\mathbf{x})$$

This can be done efficiently via dynamic programming:

$$V(t)=\underset{s=(u,t)\in s_{:t},y=(s,b),y'}{\text{argmax}}G(y,y')+V(u-1) \tag{4.5}$$

where $s_{:t}$ denotes all candidate segments ending at position t and $G(y,y')=\theta\cdot f(y,y',\mathbf{x})$. The optimal $\mathbf{y_s}^*$ can be obtained by computing $V(n)$, where n is the length of the sentence.

4.3.2 Opinion Attribute Classification

For each opinion attribute, we can define the multinomial distribution of an attribute class given a text segment. For each attribute $j \in \{1, ..., R\}$, denoting the class variable for the attribute as a^j. We have

$$P(a^j|\mathbf{x}_s) = \frac{\exp\{\phi_j \cdot g_j(a^j, \mathbf{x}_s)\}}{\sum_{a' \in \mathcal{A}_j} \exp\{\phi_j \cdot g_j(a', \mathbf{x}_s)\}} \tag{4.6}$$

where x_s denotes a text segment, ϕ_j is a parameter vector and g_j denotes feature functions for attribute a^j. The label space for polarity classification is $\{positive, negative, neutral, \varnothing\}$ and the label space for intensity classification is $\{high, medium, low, \varnothing\}$. We include an empty value $\varnothing$ to denote assigning no attribute value to those text segments that are not opinion expressions.

In the following description of our joint models, we omit the superscript on the attribute variable and derive our models with one single opinion attribute for simplicity. The derivations can be carried through with more than one opinion attribute by assuming the independence of different attributes.

4.3.3 Joint Learning Models

Joint Sequence Labeling

We can directly extend the opinion segmentation model in Section 4.3.1 to output opinion attribute labels by changing the label space to $\mathcal{Y} = \{\mathbf{y}|\mathbf{y} = \langle(s_0, \tilde{b}_0), ..., (s_k, \tilde{b}_k)\rangle\}$ where $\tilde{b}_i = (b_i, a_i) \in \{I, O\} \times \mathcal{A}$, where b_i is a binary variable as described before and a_i is an attribute class variable associated with segment s_i. Since only opinion expressions should be assigned opinion attributes, we

consider the following labeling constraints: $a_i = \varnothing$ if and only if $b_i = O$.

We can apply the same training and inference procedure described in Section 4.3.1 by replacing the label space $\mathbf{y_s}$ with the joint label space $\mathbf{y}$. Note that the feature functions are shared over the joint label space. For the loss function in the loss-augmented objective, the opinion segment label b is also replaced with the augmented label $\tilde{b}$.

Hierarchical Joint Sequence Labeling

In the above joint sequence labeling model, the opinion segmentation and attribute classification subtasks share the same set of features and parameters. In the following, we introduce an alternative approach that allows segmentation- and attribute-specific parameters and define a joint probability distribution over these parameters.

Note that the joint label space naturally forms a hierarchical structure: the process of choosing an output label $\mathbf{y}$ can be interpreted as first choosing an opinion segmentation $\mathbf{y_s} = \langle (s_0, b_0), ..., (s_k, b_k) \rangle$ and then choosing a sequence of attribute labels $\mathbf{y_a} = \langle a_0, ..., a_k \rangle$ given the chosen segment sequence. Following this intuition, the joint probability can be defined as

$$P(\mathbf{y_s}, \mathbf{y_a}|\mathbf{x}) = \frac{1}{Z_{\theta,\phi}(\mathbf{x})} \exp\left\{ \sum_{i=1}^{|\mathbf{y_s}|} \Big(\theta \cdot f(y_{s_i}, y_{s_{i-1}}, \mathbf{x}) + \phi \cdot g(a_i, y_{s_i}, \mathbf{x}) \Big) \right\}$$

where g denotes a feature function that encodes attribute-specific information for discriminating different attribute classes for each segment.

For training, we can also apply a softmax-margin by adding a loss function $l(\mathbf{y}', \mathbf{y})$ to the denominator of $P(\mathbf{y_s}, \mathbf{y_a}|\mathbf{x})$ (as in the basic joint sequence labeling

model described in Section 3.3.1).

With the estimated parameters, we can infer the optimal opinion segmentation and attribute labeling by solving

$$\underset{\mathbf{y_s},\mathbf{y_a}}{\operatorname{argmax}} \sum_{i=1}^{|\mathbf{y_s}|} \Big(\theta \cdot f(y_{s_i}, y_{s_{i-1}}, \mathbf{x}) + \phi \cdot g(a_i, y_{s_i}, \mathbf{x}) \Big) \tag{4.7}$$

We can apply a similar dynamic programming procedure as in Equation (4.5).

Our decomposition of features is similar to the hierarchical construction of CRF features in Choi and Cardie (2010). The difference is that our model is based on semi-CRFs, and the joint probability is defined over opinion segmentations and attributes of the segments. We will show that this segment-level decomposition results in better performance than the methods in Choi and Cardie (2010) in our experiments.

4.3.4 Joint Inference Models

Modeling the joint probability of opinion segmentation and attribute labeling is arguably elegant. However, training can be expensive as the computation involves normalizing over all possible segmentations and all possible attribute labelings for each segment. Thus, we also investigate joint inference approaches that combine the separately-trained models during inference.

For opinion segmentation, we train a semi-CRF-based model using the approach described in Section 4.3. For attribute classification, we train a MaxEnt model by maximizing the likelihood based on Equation (4.6). As we only need to estimate the probability of an attribute label given individual text segments, the training data can be constructed by collecting a list of text segments labeled

with correct attribute labels. The text segments do not need to form all possible sentence segmentations. To construct such training examples, we collected from each sentence all opinion expressions labeled with their corresponding attributes and use the remaining text segments as examples for the empty attribute value. The training of the MaxEnt model is much more efficient than the training of the segmentation model.

Joint Inference with Probability-based Estimates

To combine the separately-trained models at inference time, a natural inference objective is to jointly maximize the probability of opinion segmentation and the probability of attribute labeling given the chosen segmentation

$$\underset{\mathbf{y_s},\mathbf{y_a}}{\operatorname{argmax}} P(\mathbf{y_s}|\mathbf{x})P(\mathbf{y_a}|\mathbf{y_s},\mathbf{x}) \tag{4.8}$$

We can optimize it in the log space and rewrite the problem as

$$\underset{\mathbf{y_s},\mathbf{y_a}}{\operatorname{argmax}} \sum_{i=1}^{|\mathbf{y_s}|}\Big(\theta \cdot f(y_{s_i}, y_{s_{i-1}}, \mathbf{x}) + \psi(a_i, y_{s_i}, \mathbf{x})\Big) \tag{4.9}$$

where $\psi(a_i, y_{s_i}, \mathbf{x}) = \log P(a_i|\mathbf{x}_{s_i})$. Note that the optimization problem becomes very similar to Equation 4.7. In implementation, we compute $\psi(a_i, y_{s_i}, \mathbf{x}) = \alpha \log P(a_i|\mathbf{x}_{s_i})$ where $\alpha \in (0, 1]$ is a weight parameter. We found that $\alpha < 1$ provides better performance than $\alpha = 1$ empirically.

Joint Inference with Loss-based Estimates

Instead of directly combining the segmentation and classification probabilities, we explore an alternative that combines the segmentation probability with a

penalty term that penalizes attribute assignments with high loss. The joint inference objective can be written as

$$\underset{\mathbf{y_s},\mathbf{y_a}}{\operatorname{argmax}} P(\mathbf{y_s}|\mathbf{x}) \exp(-L(a_i, y_{s_i}, \mathbf{x})) \tag{4.10}$$

where $L(a_i, y_{s_i}, \mathbf{x}) = \log(E_{a_i|\mathbf{x}_{s_i}}[l(a_i, a_i')])$ is the log value of the expected loss for the predicted label a', $E_{a|\mathbf{x}_{s_i}}[l(a, a')] = \sum_a P(a|\mathbf{x}_{s_i}) l(a, a')$, and $l(a, a')$ is a loss function over a' and the true label a. We used the standard 0-1 loss function in our experiments[8]. The optimization problem can be rewritten in a similar form as Equation 4.7, and can be solved efficiently via dynamic programming.

4.3.5 Features

We consider a set of basic features as well as task-specific features for opinion segmentation and attribute labeling, respectively.

Unigrams: word unigrams and POS tag unigrams for all tokens in the segment candidate.
Bigrams: word bigrams and POS bigrams within the segment candidate.

Phrase embeddings: for each segment candidate, we associate with it a 300-dimensional phrase embedding as a dense feature representation for the segment. We make use of the recently published word embeddings trained on Google News (Mikolov et al., 2013b). For each segment, we compute the average of the word embedding vectors that comprise the phrase. We omit words that are not found in the vocabulary. If no words are found in the text segment,

[8]The loss function can be tuned to better tradeoff precision and recall according to the applications at hand. We did not explore this option in this paper.

we assign a feature vector of zeros.

Opinion lexicon: For each word in the segment candidate, we include its polarity and intensity as indicated in an existing Subjectivity Lexicon (Wilson et al., 2005b).

Segmentation-specific Features

Boundary words and POS tags: word-level features (words, POS, lexicon) before and after the segment candidate.

Phrase structure: the syntactic categories of the deepest constituents that cover the segment in the parse tree, e.g., NP, VP, TO_VB.

VP patterns: VP-related syntactic patterns described in Section 4.2.3.

Polarity-specific Features

Polarity count: counts of positive, negative and neutral words within the segment candidate according to the opinion lexicon.

Negation: an indicator for negators within the segment candidate.

Intensity-specific Features

Intensity count: counts of words with strong and weak intensity within the segment candidate according to the opinion lexicon.

Intensity dictionary: As suggested in Choi and Cardie (2010), we include features indicating whether the segment contains an intensifier (e.g., highly, really), a diminisher (e.g., little, less), a strong modal verb (e.g., must, will), and a weak

Number of Opinion Expressions		
Positive	Negative	Neutral
2170	4863	6368
High	Medium	Low
2805	5721	4875

Number of Documents	400
Number of Sentences	8241
Ave. Length of Opinion Expressions	2.86

Table 4.6: Statistics of the evaluation corpus

modal verb (e.g., may, could).

4.4 Main Experiments

4.4.1 Experimental Setup

All our experiments were conducted on the MPQA corpus (Wiebe et al., 2005). We used the same evaluation setting as in Choi and Cardie (2010), where 135 documents were used for development, and 10-fold cross-validation was performed on a different set of 400 documents. Each training fold consists of sentences labeled with opinion expression boundaries, and each expression is labeled with polarity and intensity. Table 4.6 shows some statistics of the evaluation data.

We used precision, recall and F1 as evaluation metrics for opinion expression extraction and computed them using both *proportional matching* and *binary matching* criteria.

We experimented with the following models:

(1) PIPELINE: first extracts the spans of opinion expressions using the semi-

CRF model in Section 4.3.1, and then assigns polarity and intensity to the extracted opinion expressions using MaxEnt models in Section 4.3.2. Note that the label space of the MaxEnt models does not include $\varnothing$ since they assume that all the opinion expressions extracted by the previous stage are correct.

(2) JSL: the joint sequence labeling method described in Section 4.3.3.

(3) HJSL: the hierarchical joint sequence labeling method described in Section 4.3.3.

(4) JI-PROB: the joint inference method using probability-based estimates (Equation 4.9).

(5) JI-LOSS: the joint inference method using loss-based estimates (Equation 4.10).

We also compared our results with previously published results from Choi and Cardie (2010) on the same task.

4.4.2 Implementation Details

All our models are log-linear models. We use L-BFGS with L2 regularization for training and set the regularization parameter to 1.0. We set the scaling parameter α in JI-PROB and JI-LOSS via grid search over values between 0.1 and 1 with increments of 0.1 using the development set.

We consider the same set of features described in Section 4.3.5 in all the models. For the pipeline and joint inference models where the opinion segmentator and attribute classifiers are separately trained, we employ basic features plus

segmentation-specific features in the opinion segmentator; and employ basic features plus attribute-specific features in the attribute classifiers.

4.4.3 Results

First we would like to investigate how much we can gain from using the loss-augmented training compared to using the standard training objective. Loss-augmented training can be applied to the opinion segmentation model in both the pipeline method and the joint inference methods, as well as to the joint sequence labeling approaches: JSL and HJSL (the loss function takes into account both the span overlap and the matching of attribute values). We evaluate two versions of each method: one uses loss-augmented training, and one uses standard log-loss training. Table 4.7 shows the results of opinion expression detection without evaluating their attributes. Similar trends can be observed in the results of opinion expression detection with respect to each attribute. We can see that incorporating the evaluation-metric-based loss function during training consistently improves the performance for all models in terms of the F1 measure. This confirms the effectiveness of loss-augmented training of our sequence models for opinion extraction. As a result, all following results are based on the loss-augmented version of our models.

From Table 4.7, we can see that PIPELINE provides a strong baseline for opinion expression extraction. In comparison, JSL and HJSL significantly improve precision but fail in recall, which indicates that joint sequence labeling is more conservative and precision-biased for extracting opinion expressions. HJSL significantly outperforms JSL, and this confirms the benefit of modeling

	Loss-augmented Training			Standard Training		
	P	R	F1	P	R	F1
PIPELINE	60.96	63.29	62.10	60.05	60.59	60.32
JSL	64.98†	54.60	59.29	67.09†	50.56	57.62
HJSL	**66.16***	56.77	61.05	**67.98**†	50.81	58.11
JI-PROB	50.95	**77.44***	61.32	50.06	**76.98***	60.54
JI-LOSS	63.77†	64.51†	**64.04***	64.97†	61.55†	**63.12***

Table 4.7: Performance on opinion expression extraction using proportional matching. In all tables, we use **bold** to indicate the highest score among all the methods; use $*$ to indicate statistically significant improvements ($p < 0.05$) over all the other methods under the paired-t test; use † to denote statistically significance ($p < 0.05$) over the pipeline baseline.

the dependency between opinion segmentation and attribute classification. In addition, we see that combining opinion segmentation and attribute classification without joint learning (JI-PROB and JI-LOSS) hurt precision but improves recall (vs. JSL and HJSL). JI-LOSS presents the best F1 performance and significantly outperforms the PIPELINE baseline in all evaluation metrics. This suggests that JI-LOSS provides an effective joint inference objective and is able to provide more balanced precision and recall than other joint approaches.

Table 4.8 shows the performance on opinion extraction with respect to polarity and intensity attributes. Similarly, we can see that JI-LOSS outperforms all other baselines in F1; HJSL outperforms JSL but is slightly worse than PIPELINE in F1; JI-PROB is recall-oriented and less effective than JI-LOSS.

We hypothesize that the worse performance of joint sequence labeling is due to the lack of sufficient dependencies between opinion expressions and attributes and in the training data. In many cases, the dependencies are not fully annotated. For example, the expression "fundamentally unfair and unjust" as a whole is labeled as an opinion expression with negative polarity. However,

	Positive			Negative			Neutral		
	P	R	F1	P	R	F1	P	R	F1
PIPELINE	45.26	43.07	44.04	50.59	47.91	49.11	40.98	49.30	44.57
JSL	**50.58**†	32.34	39.37	50.22	44.01	46.81	46.83†	39.81	42.85
HJSL	50.34†	37.06	42.59	53.29†	43.98	48.07	**47.29**†	43.27	45.03
JI-PROB	36.47	**47.81***	41.24	40.83	**54.40***	46.51	33.59	**59.22***	42.66
JI-LOSS	46.44†	44.58†	**45.40***	**54.88***	48.50	**51.40***	43.42†	52.02†	**47.09***
	High			Medium			Low		
	P	R	F1	P	R	F1	P	R	F1
PIPELINE	40.98	28.10	33.25	35.44	44.72	39.36	31.19	34.46	32.63
JSL	37.91	30.83†	**33.88**	**39.07**†	37.31	38.05	**40.95**†	26.71	32.24
HJSL	41.05	28.80	33.63	39.06†	39.71	39.17	40.01†	29.88	34.12
JI-PROB	34.82	**30.94**†	32.54	29.16	**50.89***	36.89	25.06	**42.99***	31.53
JI-LOSS	**46.11***	26.36	33.39	37.58†	43.58	**40.15***	33.85†	40.92†	**36.93***

Table 4.8: Performance on opinion expression extraction with attributes using proportional matching.

the sub-expression "unjust" can be also viewed as a negative expression but it is not annotated as an opinion expression in this example (as MPQA does not consider nested opinion expressions). As a result, the model would wrongly prefer an empty attribute to the expression "unjust". However, in our joint inference approaches, the attribute classification models are trained independently from the segmentation model, and the training examples for the classifiers only consist of correctly labeled expressions ("unjust" as a nested opinion expression in this example would not be considered in the training data for the attribute classifier). Therefore, the joint inference approaches do not suffer from this issue. Although joint inference does not account for task dependencies during training, the promising performance of JI-LOSS demonstrates that modeling label dependencies during inference can be more effective than the PIPELINE baseline.

In Table 4.8, we can see that the improvement of JI-LOSS is less significant in the *positive* class and the *high* class. This is due to the lack of training data in these classes. The improvement in the *medium* class is also less significant.

This may be because it is inherently harder to disambiguate *medium* from *low*. In general, we observe that extracting opinion expressions with correct intensity is a harder task than extracting opinion expressions with correct polarity.

	Extraction	Positive	Negative	Neutral	High	Medium	Low
PIPELINE	73.30	51.50	58.45	52.45	39.34	47.08	39.05
JSL	69.76	45.24	57.11	50.25	41.48†	45.88	36.49
HJSL	71.43	49.08	58.38	52.25	41.06†	46.82	38.45
JI-PROB	74.37†	50.93	58.20	54.03†	39.80	46.65	40.73†
JI-LOSS	**75.11***	**53.02***	**62.01***	**54.33**†	**41.79**†	**47.38**	**42.53***
Previous work (Choi and Cardie (2010))							
CRF-JSL	60.5	41.9	50.3	41.2	38.4	37.6	28.0
CRF-HJSL	62.0	43.1	52.8	43.1	36.3	40.9	30.7

Table 4.9: Performance on opinion expression extraction with attributes using binary matching.

Table 4.9 presents the F1 scores (due to space limit only F1 scores are reported) for all subtasks using the binary matching metric. We include the previously published results of Choi and Cardie (2010) for the same task using the same fold split and evaluation metric. CRF-JSL and CRF-HJSL are both joint sequence labeling methods based on CRFs. Different from JSL and HJSL, they perform sequence labeling at the token level instead of the segment level. We can see that both the pipeline and joint methods clearly outperform previous results in all evaluation criteria.[9] We can also see that JI-LOSS provides the best performance among all baselines.

4.4.4 Discussion

Joint vs. Pipeline We found that many errors made by the pipeline system are due to error propagation. Table 4.10 lists three examples, representing three

[9]Significance test was not conducted over the results in Choi and Cardie (2010) as we do not have their 10 fold results.

Example Sentences	Pipeline	Joint Models
It is the victim of an explosive situation $_{high}$ at the economic, ...	No opinions ×	✓
A white farmer who was shot dead Monday was the 10th to be killed.	the 10th to be killed $_{medium}$×	✓
They would " fall below minimum standards $_{medi}$ for humane $_{medium}$ treatment".	minimum standards for humane treatment $_{medium}$ ×	✓

Table 4.10: Examples of pipeline and joint model outputs.

Example Sentences	JointTrain	JointPred
The expression is undoubtedly strong and well thought out $_{high}$.	✓	well thought out $_{medium}$ ×
But the Sadc Ministerial Task Force said the election was free and fair $_{medium}$.	No opinions ×	✓
The president branded $_{high}$ as the "axis of evil" $_{high}$ in his statement ...	of evil $_{high}$ ×	✓

Table 4.11: Examples of joint learning and joint inference model outputs. The yellow color denotes neutral sentiment.

types of the propagated errors: (1) the attribute classifiers miss the prediction since the opinion expression extractor fails to identify the opinion expression; (2) the attribute classifiers assign attributes to a non-opinionated expression since it was mistakenly extracted; (3) the attribute classifiers misclassify the attributes since the boundaries of opinion expressions are not correctly determined by the opinion expression extractor. Our joint models are able to correct many of these errors, such as the examples in Table 4.10, due to the modeling of the dependency between opinion expression extraction and attribute classification.

Joint Learning vs. Joint Inference Note that JSL and HJSL both employ joint learning while JI-PROB and JI-LOSS employ joint inference. To investigate the difference between these two types of joint models, we look into the errors made by HJSL and JI-LOSS. In general, we observed that HJSL extracts many

fewer opinion expressions compared to JI-LOSS, and as a result, it presents high precision but low recall. The first two examples in Table 4.11 are cases where HJSL gains in precision and loses in recall, respectively. The last example in Table 4.11 shows an error made by HJSL but corrected by JI-LOSS. Theoretically, joint learning is more powerful than joint inference as it models the task dependencies during training. However, we only observe improvements on precision and see drops in recall. As discussed before, we hypothesize that this is due to the lack of sufficient jointly annotated data. We found that joint inference can be superior to both pipeline and joint learning, and it is also much more efficient in training. In our experiments on an Amazon EC2 instance with 64-bit processor, 4 CPUs and 15GB memory, training for the joint learning approaches took one hour for each training fold, but only 5 minutes for the joint inference approaches.

4.5 Additional Experiments

4.5.1 Reranking

Previous work (Johansson and Moschitti, 2011) showed that reranking is effective in improving the pipeline of opinion expression extraction and polarity classification. We extended their approach to handle both polarity and intensity and investigated the effect of reranking on both the pipeline and joint models. For the pipeline model, we generated 64-best (distinct) output with 4-best labeling at each pipeline stage; for the joint models, we generated 50-best (distinct) output using Viterbi-like dynamic programming. We trained the reranker using

the online Passive–Aggressive algorithm (Crammer et al., 2006) as in Johansson and Moschitti (2013b) with 100 iterations and a regularization constant $C = 0.01$. For features, we included the probability output by the base models, the polarity and intensity of each pair of extracted opinion expressions, and the word sequence and the POS sequence between the adjacent pairs of extracted opinion expressions.

	Extraction	Positive	Negative	Neutral	High	Medium	Low
PIPELINE + reranking	73.72	51.45	60.51	53.24	40.07	47.65	40.47
JSL + reranking	72.02	47.52	59.81	52.84	41.04†	46.58	39.40
HJSL + reranking	72.60	50.78	60.85	53.45	41.04†	47.75	40.08
JI-PROB + reranking	74.81†	51.45	59.59	53.98	40.66	46.87	40.80
JI-LOSS + reranking	**75.59**†	**53.29***	**62.50***	**54.94***	**41.79***	**47.67**	**42.66***

Table 4.12: Performance on opinion expression extraction with attributes using reranking and binary matching.

Table 4.12 shows the reranking performance (F1) for all subtasks. We can see that after reranking, JI-LOSS still provides the best performance and HJSL achieves comparable performance to PIPELINE. We also found that reranking leads to less performance gain for the joint inference approaches than for the joint learning approaches. This is because the k-best output of JI-PROB and JI-LOSS present less diversity than JSL and HJSL. A similar issue for reranking has also been discussed in Finkel et al. (2006).

4.5.2 Evaluation on Sentence-level Prediction Tasks

As an additional experiment, we consider a supervised sentence-level sentiment classification task using features derived from the prediction output of different opinion extraction models. As a standard baseline, we train a MaxEnt classifier using unigrams, bigrams and opinion lexicon features extracted from the sen-

tence. Using the prediction output of an opinion extraction model, we construct features by using only words from the extracted opinion expressions, and include the predicted opinion attributes as additional features. We hypothesize that the more informative the extracted opinion expressions are, the more they can contribute to sentence-level sentiment classification as features. Table 4.13 shows the results in terms of classification accuracy and F1 score in each sentiment category. BOW is the standard MaxEnt baseline. We can see that using features constructed from the opinion expressions *always* improved the performance. This confirms the informativeness of the extracted opinion expressions. In particular, using the opinion expressions extracted by JI-LOSS gives the best performance among all the baselines in all evaluation criteria. This is consistent with its superior performance in our previous experiments.

Features	Acc	Positive	Negative	Neutral
BOW	65.26	51.90	77.47	36.41
PIPELINE-OP	67.41	55.49	79.42	39.48
JSL-OP	65.86	55.97	77.68	36.46
HJSL-OP	66.79	55.12	79.29	37.56
JI-PROB-OP	67.13	56.49	79.30	38.49
JI-LOSS-OP	**68.23***	**57.32***	**80.12***	**40.45***

Table 4.13: Performance on seentence-level sentiment classification

4.6 Chapter Summary

In this chapter, we presented a semi-CRF based model for segment-level opinion expression extraction and several extensions of the model for joint opinion expression extraction and attribute classification. We showed that the semi-CRF based model can more effectively identify text spans of opinion expressions than traditional CRF based approaches; and that it can be integrated

with segment-level attribute classification in a probabilistic framework for joint opinion segmentation and attribute classification. We experimented with two types of joint models: *joint learning* — jointly estimates the segmentation- and attribute-specific parameters and *joint inference* — separately estimates the segmentation- and attribute-specific parameters and combines them only during prediction time. We showed that both types of models achieved substantially better performance than the previously published results. We also found that joint inference can be more effective and efficient than joint learning for the task. In addition, we demonstrated the usefulness of the outputs of our joint models for sentence-level prediction tasks.

CHAPTER 5

CONTEXT-AWARE SENTIMENT ANALYSIS

Discourse context is important for accurately interpreting and disambiguating sentence-level or phrase-level information. In this chapter, we explore effective ways of incorporating discourse context into fine-grained text understanding. This work was published in Yang and Cardie (2014a).

Specifically, we study the problem of sentence-level sentiment analysis. Consider for example the following sentences in a product review.

> 1. This CD was pretty relaxing. 2. But I feel like a different voice should have been used. 3. The mans voice isn't very soothing. 4. Maybe a woman's voice would have been better.

Not all sentences express the same sentiment. The first sentence expresses positive sentiment towards the overall quality of the CD. However, the following three sentences express negative sentiment towards the voice in the cd.

Typical approaches to sentence-level sentiment classification employ feature-based text classifiers which treat each sentence independently without its context. In the above example, they may be effective in identifying the positive sentiment of the first sentence due to the use of the word *relaxing*, but they could be less effective in classifying the following sentence due to the lack of explicit sentiment signals. However, if we examine the second sentence within the discourse context, we can see that the word *But* clearly indicates a sentiment transition, from positive to negative. We can also see that sentence 2, 3, and 4 all talk about the same aspect of the product —"voice", and we may expect they express consistent sentiment towards this aspect.

While there have been some attempts on utilizing discourse information in sentiment analysis, most existing work only considered discourse relations between adjacent sentences or clauses (Kanayama and Nasukawa, 2006; Zhou et al., 2011; Trivedi and Eisenstein, 2013; Lazaridou et al., 2013). Very little work explored long-distance discourse relations for sentiment analysis. One exception is Somasundaran et al. (2008), which utilized opinion target coreference annotations to constrain the polarity of sentences. However, such discourse information was incorporated as hard constraints rather than soft constraints.

In this work, we propose a structured learning method for sentence-level sentiment classification, which can (1) integrate sentiment signals both within a sentence and across multiple sentences; (2) encode lexical and discourse information as soft constraints during learning and inference; (3) make use of unlabeled data to enhance learning. Specifically, we use the conditional random field (CRF) model as the learner for sentence-level sentiment classification, and incorporate lexical and discourse information as soft constraints on the CRF posterior via Posterior Regularization (PR) (Ganchev et al., 2010). PR has been successfully applied to several structural NLP tasks (Ganchev et al., 2009; Ganchev et al., 2010; Ganchev and Das, 2013). Our work is the first to explore PR for sentiment analysis. In contrast to previous work, which mostly considered one structural constraint during learning, we design a rich set of structural constraints that model discourse context and utilize them both during learning and inference.

We evaluate our approach on the sentence-level sentiment classification task using two standard product review datasets. Experimental results show that our model outperforms state-of-the-art methods in both the supervised and

semi-supervised settings. We also show that discourse knowledge is highly useful for improving sentence-level sentiment classification.

5.1 Related Work

There has been a large amount of work on sentiment analysis at various levels of granularity (Pang and Lee, 2008). In this work, we focus on the study of sentence-level sentiment classification. Existing machine learning approaches for the task can be classified based on the use of two ideas. The first idea is to exploit sentiment signals at the sentence level by learning the relevance of sentiment and words while taking into account the context in which they occur: Nakagawa et al. (2010) uses tree-CRF to model word interactions based on dependency tree structures; Choi and Cardie (2008) applies compositional inference rules to handle polarity reversal; Socher et al. (2011b) and Socher et al. (2013) compute compositional vector representations for words and phrases and use them as features in a classifier.

The second idea is to exploit sentiment signals at the inter-sentential level. Polanyi and Zaenen (2006) argue that discourse structure is important in polarity classification. Various attempts have been made to incorporate discourse relations into sentiment analysis: Pang and Lee (2004) explored the consistency of subjectivity between neighboring sentences; Mao and Lebanon (2007),McDonald et al. (2007), and Täckström and McDonald (2011a) developed structured learning models to capture sentiment dependencies between adjacent sentences; Kanayama and Nasukawa (2006) and Zhou et al. (2011) use discourse relations to constrain two text segments to have either the same polarity or op-

posite polarities; Trivedi and Eisenstein (2013) and Lazaridou et al. (2013) encode the discourse connectors as model features in supervised classifiers. Very little work has explored long-distance discourse relations. Somasundaran et al. (2008) define opinion target relations and apply them to constrain the polarity of text segments annotated with target relations. Recently, Zhang et al. (2013) explored the use of explanatory discourse relations as soft constraints in a Markov Logic Network framework for extracting subjective text segments.

Leveraging both ideas, our approach exploits sentiment signals from both intra-sentential and inter-sentential context. It has the advantages of utilizing rich discourse knowledge at different levels of context and encoding it as soft constraints during learning and inference.

Our approach can also be applied in a semi-supervised learning setting. Compared to existing semi-supervised learning approaches for sentence-level sentiment classification (Täckström and McDonald, 2011a; Täckström and McDonald, 2011b; Qu et al., 2012), our work does not rely on a large amount of coarse-grained (document-level) labeled data, instead, distant supervision mainly comes from linguistically-motivated constraints.

5.2 Structured Learning with Posterior Constraints

In this section, we present the details of our proposed approach. We formulate the sentence-level sentiment classification task as a sequence labeling problem. The inputs to the model are sentence-segmented documents annotated with sentence-level sentiment labels (positive, negative or neutral) along with a set of unlabeled documents. During prediction, the model outputs sentiment

labels for a sequence of sentences in the test document.

In what follows, we first briefly describe the framework of Posterior Regularization. Then we introduce the posterior constraints derived based on lexical and discourse knowledge. Finally, we describe how to perform learning and inference with these constraints.

5.2.1 Posterior Regularization

PR is a framework for structured learning with constraints (Ganchev et al., 2010). In this work, we apply PR in the context of CRFs for sentence-level sentiment classification.

Denote $\mathbf{x}$ as a sequence of sentences within a document and $\mathbf{y}$ as a vector of sentiment labels associated with $\mathbf{x}$. The CRF model the following conditional probabilities:

$$p_\theta(\mathbf{y}|\mathbf{x}) = \frac{\exp(\theta \cdot f(\mathbf{x}, \mathbf{y}))}{Z_\theta(\mathbf{x})}$$

where $f(\mathbf{x}, \mathbf{y})$ are the model features, θ are the model parameters, and $Z_\theta(\mathbf{x}) = \sum_{\mathbf{y}} \exp(\theta \cdot f(\mathbf{x}, \mathbf{y}))$ is a normalization constant. The objective function for a standard CRF is to maximize the log-likelihood over a collection of labeled documents plus a regularization term:

$$\max_\theta \mathcal{L}(\theta) = \max_\theta \sum_{(\mathbf{x},\mathbf{y})} \log p_\theta(\mathbf{y}|\mathbf{x}) - \frac{||\theta||_2^2}{2\delta^2}$$

PR makes the assumption that the available labeled data is not enough for learning good model parameters, but we have a set of constraints on the pos-

terior distribution of the labels. We can define the set of desirable posterior distributions as

$$\mathcal{Q} = \{q(\mathbf{Y}) : E_q[\phi(\mathbf{X}, \mathbf{Y})] = \mathbf{b}\} \tag{5.1}$$

where ϕ is a constraint function, $\mathbf{b}$ is a vector of desired values of the expectations of the constraint functions under the distribution q [1]. Note that the distribution q is defined over a collection of unlabeled documents where the constraint functions apply, and we assume independence between documents.

The PR objective can be written as the original model objective penalized with a regularization term, which minimizes the KL-divergence between the desired model posteriors and the learned model posteriors with an L2 penalty [2] for the constraint violations.

$$\max_{\theta} \mathcal{L}(\theta) - \min_{q \in \mathcal{Q}} \{KL(q(\mathbf{Y}) || p_\theta(\mathbf{Y}|\mathbf{X})) + \beta || E_q[\phi(\mathbf{X}, \mathbf{Y})] - \mathbf{b}||_2^2\} \tag{5.2}$$

The objective can be optimized by an EM-like scheme that iteratively solves the minimization problem and the maximization problem. Solving the minimization problem is equivalent to solving its dual since the objective is convex. The dual problem is

$$\arg\max_{\lambda} \lambda \cdot \mathbf{b} - \log Z_\lambda(X) - \frac{1}{4\beta} ||\lambda||_2^2 \tag{5.3}$$

We optimize the objective function 5.2 using stochastic projected gradient, and compute the learning rate using AdaGrad (Duchi et al., 2010).

[1] In general, inequality constraints can also be used. We focus on the equality constraints since we found them to express the sentiment-relevant constraints well.

[2] Other convex functions can be used for the penalty. We use L2 norm because it works well in practice. β is a regularization constant

5.2.2 Lexical and Discourse Constraints

We develop a rich set of posterior constraints for sentence-level sentiment analysis by exploiting lexical and discourse knowledge. Specifically, we construct the lexical constraints by extracting sentiment-bearing patterns within sentences and construct the discourse-level constraints by extracting discourse relations that indicate sentiment coherence or sentiment changes both within and across sentences. Each constraint can be formulated as equality between the expectation of a constraint function value and a desired value set by prior knowledge. The equality is not strictly enforced (due to the regularization in the PR objective 5.2). Therefore, all the constraints are applied as soft constraints. Table 5.1 provides intuitive description and examples for all the constraints used in our model.

Types	Description and Examples	Inter-sentential
Lexical patterns	The sentence containing a polar lexical pattern w tends to have the polarity indicated by w. Example lexical patterns are *annoying, hate, amazing, not disappointed, no concerns, favorite, recommend*.	
Discourse Connectives (clause)	The sentence containing a discourse connective c which connects its two clauses that have opposite polarities indicated by the lexical patterns tends to have neutral sentiment. Example connectives are *while, although, though, but*.	
Discourse Connectives (sentence)	Two adjacent sentences which are connected by a discourse connective c tends to have the same polarity if c indicates a *Expansion* or *Contingency* relation, e.g., *also, for example, in fact, because* ; opposite polarities if c indicates a *Comparison* relation, e.g., *otherwise, nevertheless, however*.	✓
Coreference	The sentences which contain coreferential entities appeared as targets of opinion expressions tend to have the same polarity.	✓
Listing patterns	A series of sentences connected via a listing tend to have the same polarity.	✓
Global labels	The sentence-level polarity tends to be consistent with the document-level polarity.	✓

Table 5.1: Summarization of posterior constraints for sentence-level sentiment classification.

Lexical Patterns The existence of a polarity-carrying word alone may not correctly indicate the polarity of the sentence, as the polarity can be reversed by other polarity-reversing words. We extract lexical patterns that consist of polar words and negators [3], and apply the heuristics based on compositional semantics (Choi and Cardie, 2008) to assign a sentiment value to each pattern.

We encode the extracted lexical patterns along with their sentiment values as feature-label constraints. The constraint function can be written as

$$\phi_w(x, y) = \sum_i f_w(x_i, y_i)$$

where $f_w(x_i, y_i)$ is a feature function which has value 1 when sentence x_i contains the lexical pattern w and its sentiment label y_i equals to the expected sentiment value and has value 0 otherwise. The constraint expectation value is set to be the prior probability of associating w with its sentiment value. Note that sentences with neutral sentiment can also contain such lexical patterns. Therefore, we allow the lexical patterns to be assigned a neutral sentiment with a prior probability r_0 (we compute this value as the empirical probability of neutral sentiment in the training documents). Using the polarity indicated by lexical patterns to constrain the sentiment of sentences is quite aggressive. Therefore, we only consider lexical patterns that are strongly discriminative (many opinion words in the lexicon only indicate sentiment with weak strength). The selected lexical patterns include a handful of seed patterns (such as "pros" and "cons") and the lexical patterns that have high precision (larger then 0.9) of predicting sentiment in the training data.

Discourse Connectives. Lexical patterns can be limited in capturing contex-

[3]The polar words are identified using the MPQA lexicon and the negators are identified using a handful of seed words extended by the General Inquirer dictionary and WordNet as described in Choi and Cardie (2008).

tual information since they only look at interactions between words within an expression. To capture context at the clause or sentence level, we consider discourse connectives, which are cue phrases or words that indicate discourse relations between adjacent sentences or clauses. To identify discourse connectives, we apply a discourse tagger trained on the Penn Discourse Treebank (Prasad et al., 2008) [4] to our data. Discourse connectives are tagged with four senses: *Expansion, Contingency, Comparison, Temporal*.

Discourse connectives can operate at both intra-sentential and inter-sentential level. For example, the word "although" is often used to connect two polar clauses within a sentence, while the word "however" is often used to at the beginning of the sentence to connect two polar sentences. It is important to distinguish these two types of discourse connectives. We consider a discourse connective to be intra-sentential if it has the *Comparison* sense and connects two polar clauses with opposite polarities (determined by the lexical patterns). We construct a feature-label constraint for each intra-sentential discourse connective and set its expected sentiment value to be neutral.

Unlike the intra-sentential discourse connectives, the inter-sentential discourse connectives can indicate sentiment transitions between sentences. Intuitively, discourse connectives with the senses of *Expansion* (e.g., also, for example, furthermore) and *Contingency* (e.g., as a result, hence, because) are likely to indicate sentiment coherence; discourse connectives with the sense of *Comparison* (e.g., but, however, nevertheless) are likely to indicate sentiment changes. This intuition is reasonable, but it assumes the two sentences connected by the discourse connective are both polar sentences. In general, discourse connectives can also be used to connect non-polar (neutral) sentences. Thus, it is hard to di-

[4] `http://www.cis.upenn.edu/~epitler/discourse.html`

rectly constrain the posterior expectation for each type of sentiment transitions using inter-sentential discourse connectives.

Instead, we impose constraints on the model posteriors by reducing constraint violations. We define the following constraint function:

$$\phi_{c,s}(x, y) = \sum_i f_{c,s}(x_i, y_i, y_{i-1})$$

where c denotes a discourse connective, s indicates its sense, and $f_{c,s}$ is a penalty function that takes value 1.0 when y_i and y_{i-1} form a contradictory sentiment transition, that is, $y_i \neq_{polar} y_{i-1}$ if $s \in \{Expansion, Contingency\}$, or $y_i =_{polar} y_{i-1}$ if $s = Comparison$. The desired value for the constraint expectation is set to 0 so that the model is encouraged to have less constraint violations.

Opinion Coreference Sentences in a discourse can be linked by many types of coherence relations (Jurafsky et al., 2000). Coreference is one of the commonly used relations in written text. In this work, we explore coreference in the context of sentence-level sentiment analysis. We consider a set of polar sentences to be linked by the *opinion coreference* relation if they contain co-referring opinion-related entities. For example, the following sentences express opinions towards "the speaker phone", "The speaker phone" and "it" respectively. As these opinion targets are coreferential (referring to the same entity "the speaker phone"), they are linked by the *opinion coreference* relation [5].

> My favorite features are **the speaker phone** and the radio. **The speaker phone** is very functional. I use **it** in the car, very audible even with freeway noise.

[5] In general, the opinion-related entities include both the opinion targets and the opinion holders. In this work, we only consider the targets since we experiment with single-author product reviews. The opinion holders can be included in a similar way as the opinion targets.

Our coreference relations indicated by opinion targets overlap with the *same target* relation introduced in Somasundaran et al. (2009). The differences are: (1) we encode the coreference relations as soft constraints during learning instead of applying them as hard constraints during inference time; (2) our constraints can apply to both polar and non-polar sentences; (3) our identification of coreference relations is automatic without any fine-grained annotations for opinion targets.

To extract coreferential opinion targets, we apply Stanford's coreference system (Lee et al., 2013) to extract coreferential mentions in the document, and then apply a set of syntactic rules to identify opinion targets from the extracted mentions. The syntactic rules correspond to the shortest dependency paths between an opinion word and an extracted mention. We consider the 10 most frequent dependency paths in the training data. Example dependency paths include *nsubj*(opinion, mention), *nobj*(opinion, mention), and *amod*(mention, opinion).

For sentences connected by the opinion coreference relation, we expect their sentiment to be consistent. To encode this intuition, we define the following constraint function:

$$\phi_{coref}(x, y) = \sum_{i, ant(i)=j, j \geq 0} f_{coref}(x_i, x_j, y_i, y_j)$$

where $ant(i)$ denotes the index of the sentence which contains an antecedent target of the target mentioned in sentence i (the antecedent relations over pairs of opinion targets can be constructed using the coreference resolver), and f_{coref} is a penalty function which takes value 1.0 when the expected sentiment coherency is violated, that is, $y_i \neq_{polar} y_j$. Similar to the inter-sentential discourse connectives, modeling opinion coreference via constraint violations allows the model to handle neutral sentiment. The expected value of the constraint functions is

set to 0.

Listing Patterns Another type of coherence relations we observe in online reviews is listing, where a reviewer expresses his/her opinions by listing a series of statements followed by a sequence of numbers. For example, "1. It's smaller than the ipod mini .... 2. It has a removable battery ....". We expect sentences connected by a listing to have consistent sentiment. We implement this constraint in the same form as the coreference constraint (the antecedent assignments are constructed from the numberings).

Global Sentiment Previous studies have demonstrated the value of document-level sentiment in guiding the semi-supervised learning of sentence-level sentiment (Täckström and McDonald, 2011b; Qu et al., 2012). In this work, we also take into account this information and encode it as posterior constraints. Note that these constraints are not necessary for our model and can be applied when the document-level sentiment labels are naturally available.

Based on an analysis of the Amazon review data, we observe that sentence-level sentiment usually doesn't conflict with the document-level sentiment in terms of polarity. For example, the proportion of negative sentences in the positive documents is very small compared to the proportion of positive sentences. To encode this intuition, we define the following constraint function:

$$\phi_g(x, y) = \sum_{i}^{n} \delta(y_i \neq_{polar} g)/n$$

where $g \in \{positive, negative\}$ denotes the sentiment value of a polar document, n is the total number of sentences in x, and δ is an indicator function. We hope the expectation of the constraint function takes a small value. In our experiments, we set the expected value to be the empirical estimate of the probability of "conflicting" sentiment in polar documents using the training data.

5.2.3 Training and Inference

During training, we need to compute the constraint expectations and the feature expectations under the auxiliary distribution q at each gradient step. We can derive q by solving the dual problem in 5.3:

$$q(\mathbf{y}|\mathbf{x}) = \frac{exp(\theta \cdot f(\mathbf{x}, \mathbf{y}) + \lambda \cdot \phi(\mathbf{x}, \mathbf{y}))}{Z_{\lambda,\theta}(X)} \tag{5.4}$$

where $Z_{\lambda,\theta}(X)$ is a normalization constant. Most of our constraints can be factorized in the same way as factorizing the model features in the first-order CRF model, and we can compute the expectations under q very efficiently using the forward-backward algorithm. However, some of our discourse constraints (opinion coreference and listing) can break the tractable structure of the model. For constraints with higher-order structures, we use Gibbs Sampling (Geman and Geman, 1984) to approximate the expectations. Given a sequence $\mathbf{x}$, we sample a label $\mathbf{y}_i$ at each position i by computing the unnormalized conditional probabilities $p(\mathbf{y}_i = l|\mathbf{y}_{-i}) \propto exp(\theta \cdot f(\mathbf{x}, \mathbf{y}_i = l, \mathbf{y}_{-i}) + \lambda \cdot \phi(\mathbf{x}, \mathbf{y}_i = l, \mathbf{y}_{-i}))$ and renormalizing them. Since the possible label assignments only differ at position i, we can make the computation efficient by maintaining the structure of the coreference clusters and precomputing the constraint function for different types of violations.

During inference, we find the best label assignment by computing $\arg\max_{\mathbf{y}} q(\mathbf{y}|\mathbf{x})$. For documents where the higher-order constraints apply, we use the same Gibbs sampler as described above to infer the most likely label assignment. Otherwise, we use the Viterbi algorithm.

5.3 Experiments

5.3.1 Experimental Setup

We experimented with two product review datasets for sentence-level sentiment classification: the Customer Review (CR) data (Hu and Liu, 2004b)[6] which contains 638 reviews of 14 products such as cameras and cell phones, and the Multi-domain Amazon (MD) data from the test set of Täckström and McDonald (2011a) which contains 294 reviews from 5 different domains. As in Qu et al. (2012), we chose the books, electronics and music domains for evaluation. Each domain also comes with 33,000 extra reviews with only document-level sentiment labels.

We evaluated our method in two settings: supervised and semi-supervised. In the supervised setting, we treated the test data as unlabeled data and performed transductive learning. In the semi-supervised setting, our unlabeled data consists of both the available unlabeled data and the test data. For each domain in the MD dataset, we made use of no more than 100 unlabeled documents in which our posterior constraints apply. We adopted the evaluation schemes used in previous work: 10-fold cross validation for the CR dataset and 3-fold cross validation for the MD dataset. We also report both two-way classification (positive vs. negative) and three-way classification results (positive, negative or neutral). We use accuracy as the performance measure. In our tables, boldface numbers are statistically significant by paired t-test for $p < 0.05$ against the best baseline developed in this work [7].

[6] Available at `http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html`.

[7] Significance test was not conducted over the previous methods as we do not have their results for each fold.

5.3.2 Implementation Details

We trained our model using a CRF incorporated with the proposed posterior constraints. For the CRF features, we include the tokens, the part-of-speech tags, the prior polarities of lexical patterns indicated by the opinion lexicon and the negator lexicon, the number of positive and negative tokens and the output of the vote-flip algorithm (Choi and Cardie, 2009). In addition, we include the discourse connectives as local or transition features and the document-level sentiment labels as features (only available in the MD dataset).

We set the CRF regularization parameter $\sigma = 1$ and set the posterior regularization parameter β and γ (a trade-off parameter we introduce to balance the supervised objective and the posterior regularizer in 5.2) by using grid search [8]. For approximation inference with higher-order constraints, we perform 2000 Gibbs sampling iterations where the first 1000 iterations are burn-in iterations. To make the results more stable, we construct three Markov chains that run in parallel, and select the sample with the largest objective value.

All posterior constraints were developed using the training data on each training fold. For the MD dataset, we also used the dvd domain as additional labeled data for developing the constraints.

5.3.3 Baselines

We compared our method to five baselines: (1) CRF: CRF with the same set of model features as in our method. (2) CRF-INF: CRF augmented with inference

[8] We conducted 10-fold cross-validation on each training fold with the parameter space: β : $[0.01, 0.05, 0.1, 0.5, 1.0]$ and γ : $[0.1, 0.5, 1.0, 5.0, 10.0]$.

constraints. We can incorporate the proposed constraints (constraints derived from lexical patterns and discourse connectives) as hard constraints into CRF during inference by manually setting λ in equation 5.4 to a large value,[9]. When λ is large enough, it is equivalent to adding hard constraints to the Viterbi inference. To better understand the different effects of lexical and discourse constraints, we report results for applying only the lexical constraints (CRF-INF$_{lex}$) as well as results for applying only the discourse constraints (CRF-INF$_{disc}$). (3) PR$_{lex}$: a variant of our PR model that only applies the lexical constraints. For the three-way classification task on the MD dataset, we also implemented the following baselines: (4) VOTEFLIP: a rule-based algorithm that leverages the positive, negative and neutral cues along with the effect of negation to determine the sentence sentiment (Choi and Cardie, 2009). (5) DOCORACLE: assigns each sentence the label of its corresponding document.

5.3.4 Results

We first report results on a binary (positive or negative) sentence-level sentiment classification task. For this task, we used the supervised setting and performed transductive learning for our model. Table 5.2 shows the accuracy results. We can see that PR significantly outperforms all other baselines in both the CR dataset and the MD dataset (average accuracy across domains is reported). The poor performance of CRF-INF$_{lex}$ indicates that directly applying lexical constraints as hard constraints during inference could only hurt the performance. CRF-INF$_{disc}$ slightly outperforms CRF but the improvement is not significant. In contrast, both PR$_{lex}$ and PR significantly outperform CRF, which

[9]We set λ to 1000 for the lexical constraints and -1000 to the discourse connective constraints in the experiments

Methods	CR	MD
CRF	81.1	67.0
CRF-inf$_{lex}$	80.9	66.4
CRF-inf$_{disc}$	81.1	67.2
PR$_{lex}$	81.8	69.7
PR	**82.7**	**70.6**
Previous work		
TreeCRF (Nakagawa et al., 2010)	81.4	-
Dropout LR (Wang and Manning, 2013)	82.1	-

Table 5.2: Accuracy results (%) for supervised sentiment classification (two-way).

implies that it is much more effective to incorporate lexical and discourse constraints as posterior constraints. The superior performance of PR over PR$_{lex}$ further suggests that the proper use of discourse information can significantly improve accuracy for sentence-level sentiment classification.

We also analyzed the model's performance on a three-way sentiment classification task. By introducing the "neutral" category, the sentiment classification problem becomes harder. Table 5.3 shows the results in terms of accuracy for each domain in the MD dataset. We can see that both PR and PR$_{lex}$ significantly outperform all other baselines in all domains. The rule-based baseline VOTEFLIP gave the weakest performance because it has no prediction power on sentences with no opinion words. DOCORACLE performs much better than VOTEFLIP and performs especially well on the *Music* domain. This indicates that the document-level sentiment is a very strong indicator of the sentence-level sentiment label. For the CRF baseline and its invariants, we observe a similar performance trend as in the two-way classification task: there is nearly no performance improvement from applying the lexical and discourse-connective-based constraints during CRF inference. In contrast, both PR$_{lex}$ and PR provide substantial improvements over CRF. This confirms that encoding lexical and

	Books	**Electronics**	**Music**	**Avg**
VoteFlip	44.6	45.0	47.8	45.8
DocOracle	53.6	50.5	63.0	55.7
CRF	57.4	57.5	61.8	58.9
CRF-inf$_{lex}$	56.7	56.4	60.4	57.8
CRF-inf$_{disc}$	57.2	57.6	62.1	59.0
PR$_{lex}$	60.3	59.9	63.2	61.1
PR	**61.6**	**61.0**	**64.4**	**62.3**
Previous work				
HCRF	55.9	61.0	58.7	58.5
MEM	59.7	59.6	63.8	61.0

Table 5.3: Accuracy results (%) for semi-supervised sentiment classification (three-way).

	Books	**Electronics**	**Music**
	pos/neg/neu	pos/neg/neu	pos/neg/neu
VoteFlip	43/42/47	45/46/44	50/46/46
DocOracle	54/60/49	57/54/42	72/65/52
CRF	47/51/64	60/61/52	67/60/58
CRF-inf$_{lex}$	46/52/63	59/61/50	65/59/57
CRF-inf$_{disc}$	47/51/64	60/61/52	67/61/59
PR$_{lex}$	50/56/66	64/63/53	67/64/59
PR	52/56/68	64/66/53	69/65/60

Table 5.4: F1 scores for each sentiment category (positive, negative and neutral) for semi-supervised sentiment classification

discourse knowledge as posterior constraints allows the feature-based model to gain additional learning power for sentence-level sentiment prediction. In particular, incorporating discourse constraints leads to consistent improvements to our model. This demonstrates that our modeling of discourse information is effective and that taking into account the discourse context is important for improving sentence-level sentiment analysis. We also compare our results to the previously published results on the same dataset. HCRF (Täckström and McDonald, 2011a) and MEM (Qu et al., 2012) are two state-of-the-art semi-supervised methods for sentence-level sentiment classification. We can see that our best model PR gives the best results in most categories.

Table 5.4 shows the results in terms of F1 scores for each sentiment category (positive, negative and neutral). We can see that the PR models are able to provide improvements over all the sentiment categories compared to all the baselines in general. We observe that the DOCORACLE baseline provides very strong F1 scores on the positive and negative categories especially in the Books and Music domains, but very poor F1 on the neutral category. This is because it over-predicts the polar sentences in the polar documents, and predicts no polar sentences in the neutral documents. In contrast, our PR models provide more balanced F1 scores among all the sentiment categories. Compared to the CRF baseline and its variants, we found that the PR models can greatly improve the precision of predicting positive and negative sentences, resulting in a significant improvement on the positive/negative F1 scores. However, the improvement on the neutral category is modest. A plausible explanation is that most of our constraints focus on discriminating polar sentences. They can help reduce the errors of misclassifying polar sentences, but the model needs more constraints in order to distinguish neutral sentences from polar sentences. We plan to address this issue in future work.

5.3.5 Discussion

We analyze the errors to better understand the merits and limitations of the PR model. We found that the PR model is able to correct many CRF errors caused by the lack of labeled data. The first row in Table 5.5 shows an example of such errors. The lexical features *return* and *exchange* may be good indicators of negative sentiment for the sentence. However, with limited labeled data, the CRF learner can only associate very weak sentiment signals to

Example Sentences	CRF	PR
Example 1: ⟨neg⟩ If I could, I would like to return it or exchange for something better.⟨/neg⟩	⟨neu⟩ ×	✓
Example 2: ⟨neg⟩ Things I wasn't a fan of – the ending was to cutesy for my taste.⟨/neg⟩ ⟨neg⟩ Also, all of the side characters (particularly the mom, vee, and the teacher) were incredibly flat and stereotypical to me.⟨/neg⟩	⟨neu⟩ ⟨pos⟩ ×	✓
Example 3: ⟨neg⟩ I also have excessive noise when I talk and have phone in my pocket while walking.⟨/neg⟩ ⟨neu⟩ But other models are no better.⟨/neu⟩	⟨neg⟩ ⟨pos⟩ ×	⟨neg⟩ ⟨pos⟩ ×

Table 5.5: Examples where PR succeeds and fails to correct the mistakes of CRF

these features. In contrast, the PR model is able to associate stronger sentiment signals to these features by leveraging unlabeled data for indirect supervision. A simple lexicon-based constraint during inference time may also correct this case. However, hard-constraint baselines can hardly improve the performance in general because the contributions of different constraints are not learned, and their combination may not lead to better predictions. This is also demonstrated by the limited performance of CRF-INF in our experiments.

We also found that the discourse constraints play an important role in improving the sentiment prediction. The lexical constraints alone are often not sufficient since their coverage is limited by the sentiment lexicon and they can only constrain sentiment locally. On the contrary, discourse constraints are not dependent on sentiment lexicons, and more importantly, they can provide sentiment preferences on multiple sentences at the same time. When combining discourse constraints with features from different sentences, the PR model becomes more powerful in disambiguating sentiment. The second example in Table 5.5 shows that the PR model learned with discourse constraints correctly predicts the sentiment of two sentences where no lexical constraints apply.

However, discourse constraints are not always helpful. One reason is that they do not constrain the neutral sentiment. As a result, they could not help disambiguate neutral sentiment from polar sentiment, such as the third example in Table 5.5. This is also a problem for most of our lexical constraints. In general, it is hard to learn reliable indicators for the neutral sentiment. In the MD dataset, a neutral label may be given because the sentence contains mixed sentiment or no sentiment or it is off-topic. We plan to explore more refined constraints that can deal with the neutral sentiment in future work. Another limitation of the discourse constraints is that they could be affected by the errors of the discourse parser and the coreference resolver. A potential way to address this issue is to learn discourse constraints jointly with sentiment. We plan to study this in future research.

5.4 Chapter Summary

In this chapter, we presented a structured learning approach that can effectively leverage lexical and discourse information for sentence-level sentiment classification. We designed a rich set of structural constraints that model the discourse context at both intra-sentence and inter-sentence levels and encoded them as soft constraints on model posteriors using posterior regularization. Extensive experiments showed that our model achieved better accuracy than existing supervised and semi-supervised models for sentence-level sentiment classification. While we focused on the sentence-level task, our approach can be easily extended to handle sentiment analysis at finer levels of granularity, e.g., at the phrase or clause level.

CHAPTER 6

EVENT COREFERENCE RESOLUTION

One fundamental problem in discourse understanding is *coreference resolution*. In this chapter, we study coreference resolution in the context of events. The problem concerns determining the references of events throughout a document and across multiple documents. Event coreference resolution is critical for many real-world applications such as news summarization, financial event analysis, and social event detection.

As mentioned in Section 2.2, event coreference resolution has been relatively less explored than traditional entity coreference resolution. Achieving high performance on the task is generally more difficult. This is, in part, because events typically exhibit a more complex structure than entities: a single event can be described via multiple event mentions, and a single event mention can be associated with multiple *event arguments* that characterize the participants in the event as well as spatio-temporal information (Bejan and Harabagiu, 2010). Hence, the coreference decisions for event mentions usually require the interpretation of event mentions and their arguments in context. See, for example, Figure 6, in which five event mentions across two documents all refer to the same underlying event: *Plane bombs Yida camp*.

Most previous approaches to event coreference resolution operated by extending the supervised pairwise classification model that is widely used in entity coreference resolution (Ahn, 2006; Chen et al., 2009). In this framework, pairwise distances between event mentions are modeled via event-related features (e.g., that indicate event argument compatibility), and agglomerative clustering is applied to greedily merge event mentions into clusters. A major draw-

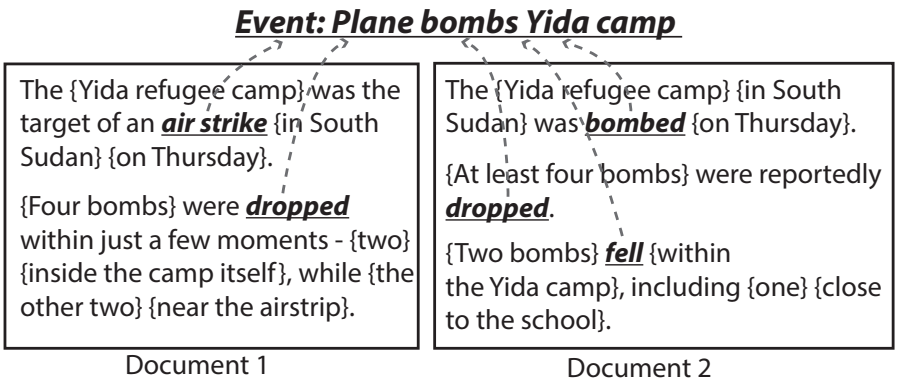

Figure 6.1: Examples of event coreference. Mutually coreferent event mentions are underlined and in boldface; participant and spatio-temporal information for the highlighted event is marked by curly brackets.

back of this general approach is that it makes hard decisions on the merging and splitting of clusters based on heuristics derived from the pairwise distances. In addition, it only captures pairwise coreference decisions within a single document and can not account for signals that commonly appear across documents. More recently, Bejan and Harabagiu (2010; 2014) proposed several nonparametric Bayesian models for event coreference resolution that probabilistically infer event clusters both within a document and across multiple documents. Their method, however, is completely unsupervised, and thus can not encode any readily available supervisory information to guide the model toward better event clustering.

To address these limitations, we propose a novel Bayesian model for within- and cross-document event coreference resolution. It leverages supervised feature-rich modeling of pairwise coreference relations and generative modeling of cluster distributions, and thus allows for both probabilistic inference over event clusters and easy incorporation of pairwise linking preferences. Our model builds on the framework of the distance-dependent Chinese restaurant

process (DDCRP) (Blei and Frazier, 2011), which was introduced to incorporate data dependencies into nonparametric clustering models. Here, however, we extend the DDCRP to allow the incorporation of feature-based, learnable distance functions as clustering priors, thus encouraging event mentions that are close in meaning to belong to the same cluster. In addition, we introduce to the DDCRP a representational hierarchy that allows event mentions to be grouped within a document and within-document event clusters to be grouped across documents.

To investigate the effectiveness of our approach, we conduct extensive experiments on the ECB+ corpus (Cybulska and Vossen, 2014b), the largest corpus available that contains event coreference annotations within and across documents. We show that integrating supervisedly-trained pairwise event coreference relations into unsupervised hierarchical modeling of event clustering achieves promising improvements over state-of-the-art approaches for both within- and cross-document event coreference resolution.

6.1 Related Work

Coreference resolution in general is a difficult natural language processing (NLP) task and typically requires sophisticated inferentially-based knowledge-intensive models (Kehler and Kehler, 2002). Extensive work in the literature focuses on the problem of entity coreference resolution and many techniques have been developed, including rule-based deterministic models (e.g. Cardie et al. (1999), Raghunathan et al. (2010), Lee et al. (2011)) that traverse over mentions in certain orderings and make deterministic coreference decisions based on all available information at the time; supervised learning-based models (e.g. Stoy-

anov et al. (2009), Rahman and Ng (2011), Durrett and Klein (2013)) that make use of rich linguistic features and the annotated corpora to learn more powerful coreference functions; and finally, unsupervised models (e.g. Bhattacharya and Getoor (2006), Haghighi and Klein (2007, 2010)) that successfully apply generative modeling to the coreference resolution problem.

Event coreference resolution is a more complex task than entity coreference resolution (Humphreys et al., 1997) and also has been relatively less studied. Existing work has adapted similar ideas to those used in entity coreference. Humphreys et al. (1997) first proposed a deterministic clustering mechanism to group event mentions of pre-specified types based on hard constraints. Later approaches (Ahn, 2006; Chen et al., 2009) applied learning-based pairwise classification decisions using event-specific features to infer event clustering. Bejan and Harabagiu (2010; 2014) proposed several unsupervised generative models for event mention clustering based on the hierarchical Dirichlet process (HDP) (Teh et al., 2006). Our approach is related to both supervised clustering and generative clustering approaches. It is a nonparametric Bayesian model in nature but encodes rich linguistic features in clustering priors. More recent work modeled both entity and event information in event coreference. Lee et al. (2012) showed that iteratively merging entity and event clusters can boost the clustering performance. Liu et al. (2014) demonstrated the benefits of propagating information between event arguments and event mentions during a post-processing step. Other work modeled event coreference as a predicate argument alignment problem between pairs of sentences, and trained classifiers for making alignment decisions (Roth and Frank, 2012; Wolfe et al., 2015). Our model also leverages event argument information into the decisions of event coreference but incorporates it into Bayesian clustering priors.

Most existing coreference models, both for events and entities, focus on solving the within-document coreference problem. Cross-document coreference has attracted less attention due to lack of annotated corpora and the requirement for larger model capacity. Hierarchical models (Singh et al., 2010; Wick et al., 2012; Haghighi and Klein, 2007) have been popular choices for cross-document coreference as they can capture coreference at multiple levels of granularities. Our model is also hierarchical, capturing both within- and cross-document coreference.

Our model is also closely related to the distance-dependent Chinese Restaurant Process (DDCRP) (Blei and Frazier, 2011). The DDCRP is an infinite clustering model that can account for data dependencies (Ghosh et al., 2011; Socher et al., 2011a). But it is a flat clustering model and thus cannot capture hierarchical structure that usually exists in large data collections. Very little work has explored the use of DDCRP in hierarchical clustering models. Kim and Oh (2011; Ghosh et al. (2011) combined a DDCRP with a standard CRP in a two-level hierarchy analogous to the HDP with restricted distance functions. Ghosh et al. (2014) proposed a two-level DDCRP with data-dependent distance-based priors at both levels. Our model is also a two-level DDCRP model but differs in that its distance function is learned using a feature-rich log-linear model. We also derive an effective Gibbs sampler for posterior inference.

6.2 Task Setup

We adopt the terminology from ECB+ (Cybulska and Vossen, 2014b), a corpus that extends the widely used EventCorefBank (ECB (Bejan and Harabagiu,

Action	*bombs*
Participant	*Sudan, Yida refugee camp*
Time	*Thursday, Nov 10, 2011*
Location	*South Sudan*

Table 6.1: Examples of event components

2010)). An **event** is something that happens or a situation that occurs (Cybulska and Vossen, 2014a). It consists of four components: (1) an *Action*: what happens in the event; (2) *Participants*: who or what is involved; (3) a *Time*: when the event happens; and (4) a *Location*: where the event happens. We assume that each document in the corpus consists of a set of mentions — text spans — that describe event actions, their participants, times, and locations. Table 6.1 shows examples of these in the sentence "Sudan bombs Yida refugee camp in South Sudan on Thursday, Nov 10th, 2011."

In this work, we also use the term **event mention** to refer to the mention of an event action, and **event arguments** to refer collectively to mentions of the participants, times, and locations involved in the event. Event mentions are usually noun phrases or verb phrases that clearly describe events. Two event mentions are considered **coreferent** if they refer to the same actual event, i.e. a situation involving a particular combination of action, participants, time and location. Note that in text, not all event arguments are always present for an event mention; they may even be distributed over different sentences. Thus, whether two event mentions are coreferential should be determined based on the context. For example, in Figure 6, the event mention *dropped* in DOCUMENT 1 corefers with *air strike* in the same document as they describe the same event, *Plane bombs Yida camp*, in the discourse context; it also corefers with *dropped* in DOCUMENT 2 based on the contexts of both documents.

To perform coreference resolution, we need to first extract all event-related mentions. We refer to this stage as *event mention extraction*. After that, we want to group event mentions into clusters according to their coreference relations. This can be done both within a document and across multiple documents. We refer to this stage as *event clustering*.

6.2.1 Event Mention Extraction

The goal of event mention extraction is to extract from a text all event mentions (actions) and event arguments (the associated participants, times, and locations). One might expect that event actions could be extracted reasonably well by identifying verb groups; and event arguments, by applying semantic role labeling (SRL) to identify, for example, the *Agent* and *Patient* of each predicate. Unfortunately, most SRL systems only handle verbal predicates and so would miss event mentions described via noun phrases. In addition, SRL systems are not designed to capture event-specific arguments. Accordingly, we found that a state-of-the-art SRL system (SwiRL (Surdeanu et al., 2007)) extracted only 56% of the actions, 76% of participants, 65% of times and 13% of locations for events in a development set of ECB+ based on a head word matching evaluation measure. (We provide dataset details in Section 6.4.) To produce higher recall, we need a learning-based event extractor that can make use of existing annotation for event actions, participants, times, and locations.

As described in Section 1.1.1, event extraction shares a lot of similarities with opinion extraction. The key elements of opinions and events can both be extracted in the form of text entities as well as their relations and attributes. Based

on this observation, we adapt the machine learning techniques we developed for extracting fine-grained opinion elements to extract event mentions and event arguments.

Specifically, we formulate the identification of text spans of event mentions and event arguments as a sequence labeling problem and apply the semi-CRF-based model described in Section 4.2 to detect the mention boundaries. We make use of a rich feature set that includes word-level features such as unigrams, bigrams, POS tags, WordNet hypernyms, synonyms and FrameNet semantic roles, and phrase-level features such as phrasal syntax (e.g., NP, VP) and phrasal embeddings (constructed by averaging word embeddings produced by word2vec (Mikolov et al., 2013a)). Our experiments on the same (held-out) development data show that the semi-CRF-based extractor correctly identifies 95% of actions, 90% of participants, 94% of times and 74% of locations again based on head word matching.

Besides identifying the boundaries of event mentions and event arguments, we need to identifying the association relations among event mentions and arguments. This can be formulated as a relation classification problem, and it can be well integrated with the mention boundary extraction problem using the joint inference framework described in Section 4.8. Specifically, in the joint objective function (3.1), the entity label z can take values from *action, participant, time, location* and the confidence scores can be obtained from the semi-CRF-based mention extractor; correspondingly, there are three types of relations that link an event action to its participants, its times, and its locations respectively. The confidence scores for the relation decisions can be obtained from supervised relation classifiers.

Due to the lack of supervisory data for event relations in the ECB+ corpus, we resort to a simple heuristic method for identifying event relations. We assume that all the event arguments identified by the semi-CRF extractor are related to all event mentions in the same sentence and then apply SRL-based heuristics to augment and further disambiguate intra-sentential action-argument relations (using the SwiRL SRL). More specifically, we link each verbal event mention to the participants that match its *ARG0*, *ARG1* or *ARG2* semantic role fillers; similarly, we associate with the event mention the time and locations that match its *AM-TMP* and *AM-LOC* role fillers, respectively. For each nominal event mention, we associate those participants that match the possessor of the mention since these were suggested in Lee et al. (2012) as playing the *ARG0* role for nominal predicates.

6.3 A Bayesian Model for Event Clustering

In this section, we describe a novel Bayesian model for event clustering. Our model is a hierarchical extension of the distance-dependent Chinese Restaurant Process (DDCRP). It first groups event mentions within a document to form within-document event cluster and then groups these event clusters across documents to form global clusters. The model can account for the similarity between event mentions during the clustering process, putting a bias toward clusters comprised of event mentions that are similar to each other based on the context. To capture event similarity, we use a log-linear model with rich syntactic and semantic features and learn the feature weights using gold-standard data.

6.3.1 Distance-dependent Chinese Restaurant Process

The Distance-dependent Chinese Restaurant Process (DDCRP) is a generalization of the Chinese Restaurant process (CRP) that models distributions over partitions. In a CRP, the generative process can be described by imagining data points as customers in a restaurant and the partitioning of data as tables at which the customers sit. The process randomly samples the table assignment for each customer sequentially: the probability of a customer sitting at an existing table is proportional to the number of customers already sitting at that table and the probability of sitting at a new table is proportional to a scaling parameter. For each customer sitting at the same table, an observation can be drawn from a distribution determined by the parameter associated with that table. Despite the sequential sampling process, the CRP makes the assumption of exchangeability: the permutation of the customer ordering does not change the probability of the partitions.

The exchangeability assumption may not be reasonable for clustering data that has clear inter-dependencies. The DDCRP allows the incorporation of data dependencies in infinite clustering, encouraging data points that are closer to each other to be grouped together. In the generative process, instead of directly sampling a table assignment for each customer, it samples a customer link, linking the customer to another customer or itself. The clustering can be uniquely constructed once the customer links are determined for all customers: two customers belong to the same cluster if and only if one can reach the other by traversing the customer links (treating these links as undirected).

More formally, consider a sequence of customers $1, ..., n$, and denote $\mathbf{a}$ =

$(a_1, ..., a_n)$ as the assignments of the customer links. $a_i \in \{1, \ldots, n\}$ is drawn from

$$p(a_i = j|F, \alpha) \propto \begin{cases} F(i, j), & j \neq i \\ \alpha, & j = i \end{cases} \tag{6.1}$$

where F is a distance function and $F(i, j)$ is a value that measures the distance between customer i and j. α is a scaling parameter, measuring self-affinity. For each customer, the observation is generated by the per-table parameters as in the CRP. A DDCRP is said to be *sequential* if $F(i, j) = 0$ when $i < j$, so customers may link only to themselves, and to previous customers.

6.3.2 A Hierarchical Extension of the DDCRP

We can model within-document coreference resolution using a sequential DDCRP. Imagining customers as event mentions and the restaurant as a document, each mention can either refer to an antecedent mention in the document or no other mentions, starting the description of a new event. However, the coreference relations may also exist across documents — the same event may be described in multiple documents. Thus it is ideal to have a two-level clustering model that can group event mentions within a document and further group them across documents. Therefore we propose a hierarchical extension of the DDCRP (HDDCRP) that employs a DDCRP twice: the first-level DDCRP links mentions based on within-document distances and the-second level DDCRP links the within-document clusters based on cross-document distances, forming larger clusters in the corpus.

The generative process of an HDDCRP can be described using the same "Chinese Restaurant" metaphor. Imagine a collection of documents as a col-

lection of restaurants, and the event mentions in each document as customers entering a restaurant. The local (within-document) event clusters correspond to *tables*. The global (within-corpus) event clusters correspond to *menus* (tables that serve the same menu belong to the same cluster). The hidden variables are the customer links and the table links. Figure 6.2 shows a configuration of these variables and the corresponding clustering structure.

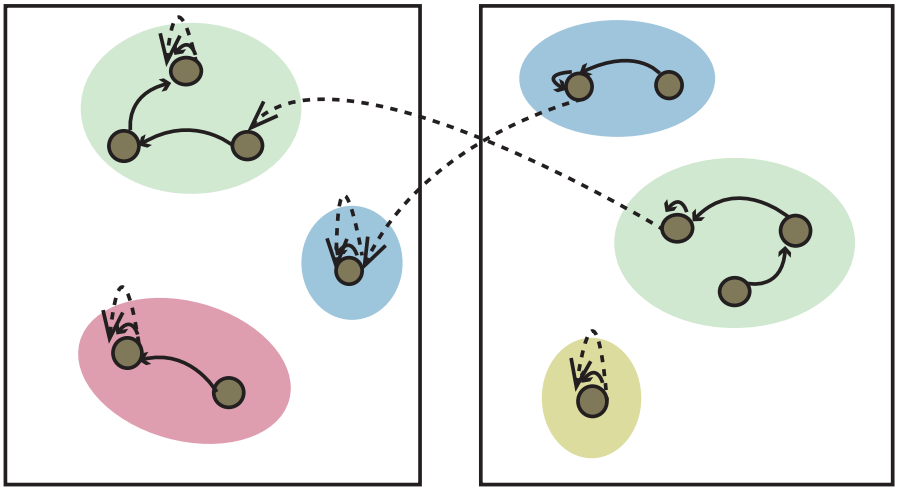

Figure 6.2: A cluster configuration generated by the HDDCRP. Each restaurant is represented by a rectangle. The small green circles represent customers. The ovals represent tables and the colors reflect the clustering. Each customer is assigned a customer link (a solid arrow), linking to itself or another customer in the same restaurant. The customer who first sits at the table is assigned a table link (a dashed arrow), linking to itself or another customer in a different restaurant, resulting in the linking of two tables.

More formally, the generative process for the HDDCRP can be described as follows:

1. For each restaurant $d \in \{1, ..., D\}$, for each customer $i \in \{1, ..., n_d\}$, sample a

customer link using a sequential DDCRP:

$$p(a_{i,d} = (j,d)) \propto \begin{cases} F_d(i,j), & j < i \\ \alpha_d, & j = i \\ 0, & j > i \end{cases} \tag{6.2}$$

2. For each restaurant $d \in \{1, ..., D\}$, for each table t, sample a table link for the customer (i,d) who first sits at t using a DDCRP:

$$p(c_{i,d} = (j,d')) \propto \begin{cases} F_0((i,d),(j,d')), & j \in \{1, ..., n_{d'}\}, d' \neq d \\ \alpha_0, & j = i, d' = d \end{cases} \tag{6.3}$$

3. Calculate clusters $\mathbf{z}(\mathbf{a}, \mathbf{c})$ by traversing all the customer links $\mathbf{a}$ and the table links $\mathbf{c}$. Two customers are in the same cluster if and only if there is a path from one to the other along the links, where we treat both table and customer links as undirected.
4. For each cluster $k \in \mathbf{z}(\mathbf{a}, \mathbf{c})$, sample parameters $\phi_k \sim G_0(\lambda)$.
5. For each customer i in cluster k, sample an observation $x_i \sim p(\cdot|\phi_{z_i})$ where $z_i = k$.

$F_{1:D}$ and F_0 are distance functions that map a pair of customers to a distance value. We will discuss them in detail in Section 6.3.4.

6.3.3 Posterior Inference with Gibbs Sampling

The central computation problem for the HDDCRP model is posterior inference — computing the conditional distribution of the hidden variables given the ob-

servations $p(\mathbf{a}, \mathbf{c}|\mathbf{x}, \alpha_0, F_0, \alpha_{1:D}, F_{1:D})$. The posterior is intractable due to a combinatorial number of possible link configurations. Thus we approximate the posterior using Markov Chain Monte Carlo (MCMC) sampling, and specifically using a Gibbs sampler.

In developing this Gibbs sampler, we first observe that the generative process is equivalent to one that, in step 2 samples a table link for *all* customers, and then in step 3, when calculating $\mathbf{z}(\mathbf{a}, \mathbf{c})$, includes only those table links $c_{i,d}$ originating at customers (i, d) that started a new table, i.e. that chose $a_{i,d} = (i, d)$.

The Gibbs sampler for the HDDCRP iteratively samples a customer link for each customer (i, d) from

$$p(a^*_{i,d}|\mathbf{a}_{-(i,d)}, \mathbf{c}, \mathbf{x}, \lambda) \propto p(a^*_{i,d}) H_a(\mathbf{x}, \mathbf{z}, \lambda) \tag{6.4}$$

where

$$H_a(\mathbf{x}, \mathbf{z}, \lambda) = \frac{p(\mathbf{x}|\mathbf{z}(\mathbf{a}_{-(i,d)} \cup a^*_{i,d}, \mathbf{c}, \lambda))}{p(\mathbf{x}|\mathbf{z}(\mathbf{a}_{-(i,d)}, \mathbf{c}), \lambda))}$$

After sampling all the customer links, it samples a table link for all customers (i, d) according to

$$p(c^*_{i,d}|\mathbf{a}, \mathbf{c}_{-(i,d)}, \mathbf{x}, \lambda) \propto p(c^*_{i,d}) H_c(\mathbf{x}, \mathbf{z}, \lambda) \tag{6.5}$$

where

$$H_c(\mathbf{x}, \mathbf{z}, \lambda) = \frac{p(\mathbf{x}|\mathbf{z}(\mathbf{a}, \mathbf{c}_{-(i,d)} \cup c^*_{i,d}, \lambda))}{p(\mathbf{x}|\mathbf{z}(\mathbf{a}, \mathbf{c}_{-(i,d)}), \lambda))}$$

For those customers (i, d) that did not start a new table, i.e. with $a_{i,d} \neq (i, d)$, the table link $c^*_{i,d}$ does not affect the clustering, and so $H_c(\mathbf{x}, \mathbf{z}, \lambda) = 1$ in this case.

Referring back to the event coreference example in 6, Figure 6.3 shows an example of variable configuration for the HDDCRP model and the corresponding coreference clusters.

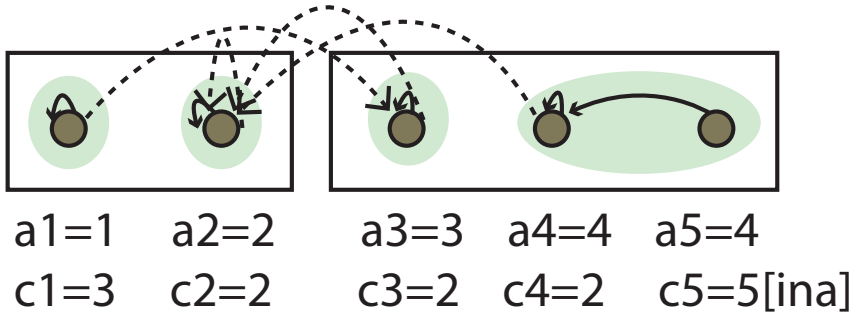

Figure 6.3: An example of event clustering and the corresponding variable assignments. The assignments of **a** induce tables, or within-document (WD) clusters, and the assignments of **c** induce menus, or cross-document (CD) clusters. [ina] denotes that the variable is inactive and will not affect the clustering.

In implementation, we can simplify the computations of both $H_a(\mathbf{x}, \mathbf{z}, \lambda)$ and $H_c(\mathbf{x}, \mathbf{z}, \lambda)$ by using the fact that the likelihood under clustering $\mathbf{z}(\mathbf{a}, \mathbf{c})$ can be factorized as

$$p(\mathbf{x}|\mathbf{z}(\mathbf{a}, \mathbf{c}), \lambda) = \prod_{k \in \mathbf{z}(\mathbf{a},\mathbf{c})} p(\mathbf{x}_{\mathbf{z}=k}|\lambda)$$

where $\mathbf{x}_{\mathbf{z}=k}$ denotes all customers that belong to the global cluster k. $p(\mathbf{x}_{\mathbf{z}=k}|\lambda)$ is the marginal probability. It can be computed as

$$p(\mathbf{x}_{\mathbf{z}=k}|\lambda) = \int p(\phi|\lambda) \prod_{i \in \mathbf{z}=k} p(x_i|\phi) d\phi$$

where x_i is the observation associated with customer i. In our problem, the observation corresponds to the lemmatized words in the event mention. We model the observed word counts using cluster-specific multinomial distributions with symmetric Dirichlet priors.

6.3.4 Feature-based Distance Functions

The distance functions $F_{1:D}$ and F_0 encode the priors for the clustering distribution, preferring clustering data points that are closer to each other. We consider

event mentions as the data points and encode the similarity (or compatibility) between event mentions as priors for event clustering. Specifically, we use a log-linear model to estimate the similarity between a pair of event mentions (x_i, x_j)

$$f_{\boldsymbol{\theta}}(x_i, x_j) \propto \exp\{\boldsymbol{\theta}^T \psi(x_i, x_j)\} \tag{6.6}$$

where ψ is a feature vector, containing a rich set of features based on event mentions i and j: (1) head word string match, (2) head POS pair, (3) cosine similarity between the head word embeddings (we use the pre-trained 300-dimensional word embeddings from word2vec[1]), (4) similarity between the words in the event mentions (based on term frequency (TF) vectors), (5) the Jaccard coefficient between the WordNet synonyms of the head words, and (6) similarity between the context words (a window of three words before and after each event mention). If both event mentions involve participants, we consider the similarity between the words in the participant mentions based on the TF vectors, similarly for the time mentions and the location mentions. If the SRL role information is available, we also consider the similarity between words in each SRL role, i.e. Arg0, Arg1, Arg2.

Training We train the parameter $\boldsymbol{\theta}$ using logistic regression with an L2 regularizer. We construct the training data by considering all ordered pairs of event mentions within a document, and also all pairs of event mentions across similar documents. To measure document similarity, we collect all mentions of events, participants, times and locations in each document and compute the cosine similarity between the TF vectors constructed from all the event-related mentions. We consider two documents to be similar if their TF-based similarity is above a threshold σ (we set it to 0.4 in our experiments).

[1] `https://code.google.com/p/word2vec/`

After learning $\boldsymbol{\theta}$, we set the within-document distances as $F_d(i, j) = f_{\boldsymbol{\theta}}(x_i, x_j)$, and the across-document distances as $F_0((i, d), (j, d')) = w(d, d') f_{\boldsymbol{\theta}}(x_{i,d}, x_{j,d'})$, where $w(d, d') = \exp(\gamma sim(d, d'))$ captures document similarity where $sim(d, d')$ is the TF-based similarity between document d and d', and γ is a weight parameter. Higher γ leads to a higher effect of document-level similarities on the linking probabilities. We set $\gamma = 1$ in our experiments.

6.4 Experiments

6.4.1 Experimental Setup

We conduct experiments using the ECB+ corpus (Cybulska and Vossen, 2014b), the largest available dataset with annotations of both within-document (WD) and cross-document (CD) event coreference resolution. It extends ECB 0.1 (Lee et al., 2012) and ECB (Bejan and Harabagiu, 2010) by adding event argument and argument type annotations as well as adding more news documents. The cross-document coreference annotations only exist in documents that describe the same seminal event (the event that triggers the topic of the document and has interconnections with the majority of events from its surrounding textual context (Bejan and Harabagiu, 2014)). We divide the dataset into a training set (topics 1-20), a development set (topics 21-23), and a test set (topics 24-43). Table 6.2 shows the statistics of the data.

We performed event coreference resolution on all possible event mentions that are expressed in the documents. Using the event extraction method described in Section 6.2.1, we extracted 53,429 event mentions, 43,682 participant

	Train	Dev	Test	Total
# Documents	462	73	447	982
# Sentences	7,294	649	7,867	15,810
# Annotated event mentions	3,555	441	3,290	7,286
# Cross-document chains	687	47	486	1,220
# Within-document chains	2,499	316	2,137	4,952

Table 6.2: Statistics of the ECB+ corpus

mentions, 5,791 time mentions and 3,836 location mentions in the test data, covering 93.5%, 89.0%, 95.0%, 72.8% of the annotated event mentions, participants, time and locations, respectively.

We evaluate both within- and cross-document event coreference resolution. As in previous work (Bejan and Harabagiu, 2010), we evaluate cross-document coreference resolution by merging all documents from the same seminal event into a meta-document and then evaluate the meta-document as in within-document coreference resolution. However, during inference time, we do not assume the knowledge of the mapping of documents to seminal events.

We consider three widely used coreference resolution metrics: (1) MUC (Vilain et al., 1995), which measures how many gold (predicted) cluster merging operations are needed to recover each predicted (gold) cluster; (2) B^3 (Bagga and Baldwin, 1998), which measures the proportion of overlap between the predicted and gold clusters for each mention and computes the average scores; and (3) CEAF (Luo, 2005) ($CEAF_e$), which measures the best alignment of the gold-standard and predicted clusters. We also consider the CoNLL F1, which is the average F1 of the above three measures. All the scores are computed using the latest version (v8.01) of the official CoNLL scorer (Pradhan et al., 2014).

6.4.2 Baselines

We compare our proposed HDDCRP model (HDDCRP) to five baselines:

- LEMMA: a heuristic method that groups all event mentions, either within or across documents, which have the same lemmatized head word. It is usually considered a strong baseline for event coreference resolution.
- AGGLOMERATIVE: a supervised clustering method for within-document event coreference (Chen et al., 2009). We extend it to within- and cross-document event coreference by performing single-link clustering in two phases: first grouping mentions within documents and then grouping within-document clusters to larger clusters across documents. We compute the pairwise-linkage scores using the log-linear model described in Section 6.3.4.
- HDP-LEX: an unsupervised Bayesian clustering model for within- and cross-document event coreference (Bejan and Harabagiu, 2010)[2]. It is a hierarchical Dirichlet process (HDP) model with the likelihood of all the lemmatized words observed in the event mentions. In general, the HDP can be formulated using a two-level sequential CRP. Our HDDCRP model is a two-level DDCRP that generalizes the HDP to allow data dependencies to be incorporated at both levels[3].

[2]We re-implement the proposed HDP-based models: the HDP_{1f}, HDP_{flat} (including HDP_{flat} (LF), (LF+WF), and (LF+WF+SF)) and HDP_{struct}, but found that the HDP_{flat} with lexical features (LF) performs the best in our experiments. We refer to it as HDP-LEX.

[3]Note that HDP-LEX is not a special case of HDDCRP because we define the table-level distance function as the distances between customers instead of between tables. In our model, the probability of linking a table t to another table s depends on the distance between the head customer at table t and all other customers who sit at table s. Defining the table-level distance function this way allows us to derive a tractable inference algorithm using Gibbs sampling.

- DDCRP: a DDCRP model we develop for event coreference resolution. It applies the distance prior in Equation 6.1 to all pairs of event mentions in the corpus, ignoring the document boundaries. It uses the same likelihood function and the same log-linear model to learn the distance values as HDDCRP. But it has fewer link variables than HDDCRP and it does not distinguish between the within-document and cross-document link variables. For the same clustering structure, HDDCRP can generate more possible link configurations than DDCRP.

- HDDCRP*: a variant of the proposed HDDCRP that only incorporates the within-document dependencies but not the cross-document dependencies. The generative process of HDDCRP* is similar to the one described in Section 6.3.2, except that in step 2, for each table t, we sample a cluster assignment c_t according to

$$p(c_t = k) \propto \begin{cases} n_k, & k \leq K \\ \alpha_0, & k = K + 1 \end{cases}$$

  where K is the number of existing clusters, n_k is the number of existing tables that belong to cluster k, α is the concentration parameter. And in step 3, the clusters $\mathbf{z}(\mathbf{a}, \mathbf{c})$ are constructed by traversing the customer links and looking up the cluster assignments for the obtained tables. We also use Gibbs sampling for inference.

6.4.3 Parameter settings

For all the Bayesian models, the reported results are averaged results over five MCMC runs, each for 500 iterations. We found that mixing happens before 500

iterations in all models by observing the joint log-likelihood. For the DDCRP, HDDCRP* and HDDCRP, we randomly initialized the link variables. Before initialization, we assume that each mention belongs to its own cluster. We assume mentions are ordered according to their appearance within a document, but we do not assume any particular ordering of documents. We also truncated the pairwise mention similarity to zero if it is below 0.5 as we found that it leads to better performance on the development set. We set $\alpha_1 = ... = \alpha_D = 0.5$, $\alpha_0 = 0.001$ for HDDCRP, $\alpha_0 = 1$ for HDDCRP*, $\alpha = 0.1$ for DDCRP, and $\lambda = 10^{-7}$. All the hyperparameters were set based on the development data.

6.4.4 Main Results

Table 6.3 shows the event coreference results. We can see that LEMMA-matching is a strong baseline for event coreference resolution. HDP-LEX provides noticeable improvements, suggesting the benefit of using an infinite mixture model for event clustering. AGGLOMERATIVE further improves the performance over HDP-LEX for WD resolution, however, it fails to improve CD resolution. We conjecture that this is due to the combination of ineffective thresholding and the prediction errors on the pairwise distances between mention pairs across documents. Overall, HDDCRP* outperforms all the baselines in CoNLL F1 for both WD and CD evaluation. The clear performance gains over HDP-LEX demonstrate that it is important to account for pairwise mention dependencies in the generative modeling of event clustering. The improvements over AGGLOMERATIVE indicate that it is more effective to model mention-pair dependencies as clustering priors than as heuristics for deterministic clustering.

	MUC			B^3			$CEAF_e$			CoNLL
	P	R	F1	P	R	F1	P	R	F1	F1
	Cross-document Event Coreference Resolution (CD)									
LEMMA	75.1	55.4	63.8	71.7	39.6	51.0	36.2	61.1	45.5	53.4
HDP-LEX	75.5	63.5	69.0	65.6	43.7	52.5	34.8	60.2	44.1	55.2
AGGLOMERATIVE	78.3	59.2	67.4	73.2	40.2	51.9	30.2	65.6	41.4	53.6
DDCRP	79.6	58.2	67.1	78.1	39.6	52.6	31.8	**69.4**	43.6	54.4
HDDCRP*	77.5	66.4	71.5	69.0	**48.1**	**56.7**	38.2	63.0	47.6	58.6
HDDCRP	**80.3**	**67.1**	**73.1**	**78.5**	40.6	53.5	**38.6**	68.9	**49.5**	**58.7**
	Within-document Event Coreference Resolution (WD)									
LEMMA	60.9	30.2	40.4	78.9	57.3	66.4	63.6	69.0	66.2	57.7
HDP-LEX	50.0	39.1	43.9	74.7	67.6	71.0	66.2	71.4	68.7	61.2
AGGLOMERATIVE	61.9	39.2	48.0	80.7	67.6	73.5	65.6	76.0	70.4	63.9
DDCRP	71.2	36.4	48.2	85.4	64.9	73.8	61.8	76.1	68.2	63.4
HDDCRP*	58.1	**42.8**	49.3	78.4	**68.7**	73.2	**67.6**	74.5	70.9	64.5
HDDCRP	**74.3**	41.7	**53.4**	**85.6**	67.3	**75.4**	65.1	**79.8**	**71.7**	**66.8**

Table 6.3: Within- and cross-document coreference results on the ECB+ corpus

Comparing among the HDDCRP-related models, we can see that HDDCRP clearly outperforms DDCRP, demonstrating the benefits of incorporating the hierarchy into the model. HDDCRP also performs better than HDDCRP* in WD CoNLL F1, indicating that incorporating cross-document information helps within-document clustering. We can also see that HDDCRP performs similarly to HDDCRP* in CD CoNLL F1 due to the lower B^3 F1, in particular, the decrease in B^3 recall. This is because applying the DDCRP prior at both within- and cross-document levels results in more conservative clustering and produces smaller clusters. This could be potentially improved by employing more accurate similarity priors.

To further understand the effect of modeling mention-pair dependencies, we analyze the impact of the features in the mention-pair similarity model. Table 6.4 lists the learned weights of some top features (sorted by weights). We

Features	Weight
Head Embedding sim	4.5
String match	2.77
Context sim	1.75
Synonym sim	1.56
TF sim	1.17
SRL Arg1 sim	1.10
SRL Arg0 sim	0.89
Participant sim	0.68

Table 6.4: Learned weights for selected features

can see that they mainly serve to discriminate event mentions based on the head word similarity (especially embedding-based similarity) and the context word similarity. Event argument information such as *SRL Arg1*, *SRL Arg0*, and *Participant* are also indicative of the coreferential relations.

6.4.5 Discussion

We found that HDDCRP corrects many errors made by the traditional agglomerative clustering model (AGGLOMERATIVE) and the unsupervised generative model (HDP-LEX). AGGLOMERATIVE easily suffers from error propagation as the errors made by the supervised distance learner cannot be corrected. HDP-LEX often mistakenly groups mentions together based on word co-occurrence statistics but not the apparent similarity features in the mentions. In contrast, HDDCRP avoids such errors by performing probabilistic modeling of clustering and making use of rich linguistic features trained on available annotated data. For example, HDDCRP correctly groups the event mention "unveiled" in *"Apple's Phil Schiller unveiled a revamped MacBook Pro today"* together with the event mention "announced" in *"this notebook isn't the only laptop Apple announced for the MacBook Pro lineup today"*, while both HDP-LEX and AGGLOMERATIVE models

fail to make such connection.

By looking further into the errors, we found that a lot of mistakes made by HDDCRP are due to the errors in event extraction and pairwise linkage prediction. The event extraction errors include false positive and false negative event mentions and event arguments, boundary errors for the extracted mentions, and argument association errors. The pairwise linking errors often come from the lack of semantic and world knowledge, and this applies to both event mentions and event arguments, especially for time and location arguments which are less likely to be repeatedly mentioned and in many cases require external knowledge to resolve their meanings, e.g., "*May 3, 2013*" is "*Friday*" and "*Mount Cook*" is "*New Zealand's highest peak*".

6.5 Chapter Summary

In this chapter, we presented a model that can perform Bayesian clustering with feature-rich priors for event coreference resolution, both within- and cross-document. Our model leverages the advantages of generative modeling of coreference resolution and feature-based discriminative modeling of mention reference relations. We showed its power in resolving event coreference by comparing it to a traditional pairwise clustering approach and a state-of-the-art unsupervised generative clustering approach. It is worth noting that our clustering model is general and can be applied to cluster any groups of objects that exhibit rich pairwise compatibility properties.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In this dissertation, we presented computational models that push the envelope of automatic extraction of opinions and events from text. An overarching theme in these models is joint inference: simultaneously considering multiple sources of low-level information and aggregating them across different parts of text.

7.1 Summary of Contributions

In Chapter 3, we presented an integer linear programming based approach for joint extraction of opinion expressions, holders, targets, together with their relations. We demonstrated that simultaneously considering all these elements improved performance on the extraction of each individual element, and significantly outperformed the state-of-the-art approaches for the task.

Chapter 4 presented joint models for opinion expression extraction and opinion attribute classification. Standard approaches to the task identify the text spans of opinion expressions first and then assign attribute labels (in particular, polarity and intensity) to each extracted opinion expression. We presented several alternatives to such pipeline approaches. We modeled the joint distribution over the segmentation of opinion expressions and the classification of opinion attributes by defining segmentation- and attribute-specific potential functions. We explored two types of joint models: one estimates the segmentation- and attribute-specific parameters jointly during training and the other estimates the parameters separately and combine them only during inference. We found that

joint inference is more effective for the task due to training efficiency and effective use of existing annotated data. It also provides significant improvements over the standard pipeline approach as well as a state-of-the-art reranking approach (reranking the k-best outputs from each stage in the pipeline).

Chapter 5 explored learning techniques that allow effective integration of discourse context for accurate interpretation of fine-grained information. We focused on the task of sentence-level sentiment classification. Existing approaches to the task mostly treat sentences independently and make predictions only based on information within each sentence. We presented a CRF-based model that can effectively utilize information about relations between neighboring sentences or long-distance sentences, and encode it as soft structural constraints on CRF using posterior regularization (PR) (Ganchev et al., 2010). Extensive experiments showed that our model demonstrated promising improvements over the standard CRFs and models that simply utilized the inter-sentential cues as hard constraints on the CRF outputs. Furthermore, our model can improve performance in a semi-supervised learning setting where unlabeled data is utilized to provide distant supervision during learning.

Finally, chapter 6 studied coreference resolution, the problem that is fundamental for a discourse-level understanding of text. We focused on coreference resolution of events, both within a document and across multiple documents. We first showed how to apply similar extraction methods for opinions to the extraction of event-related mentions. Then, we proposed a novel Bayesian clustering model for clustering event mentions within a document and across documents. Unlike conventional coreference resolution models, our model performs Bayesian inference of the cluster distributions while allowing for feature-rich

priors that capture pairwise coreference relations. By leveraging the pairwise coreference priors and the global clustering likelihood, our model significantly outperformed a standard pairwise clustering approach and a state-of-the-art Bayesian approach for both within- and cross-document event coreference resolution.

7.2 Future Work

Our work in this dissertation has demonstrated that combining modern machine learning techniques with insights to language understanding can significantly improve automatic extraction of complex information. This opens many new opportunities for future research. In the following, we discuss future work in several directions:

A general ILP-based framework for information extraction. Ideally, we want to build a unified system that can extract different kinds of information, including opinions, events, and other types of semantic information. Suppose we can represent different types of information using similar definitions of entities, relations, and attributes. We can easily generalize the ILP-based framework in Section 3 for joint entity, attribute and relation extraction. Specifically, the ILP objective can be written as:

$$\arg\max_{x,t,u,v} \sum_{i\in\mathcal{S}} \left(\sum_{z} f_{iz} x_{iz} + \sum_{a} g_{ia} t_{ia} \right) + \sum_{k} \sum_{i\in\mathcal{S}} \left(\sum_{j\in\mathcal{A}_k} r_{ij} u_{ij} + r_{i\emptyset} v_{ik} \right) \tag{7.1}$$

where the first summation term optimizes the entity and attribute assignments to candidate text spans, and the second summation term optimizes the relation

assignments. Similar to objective function 3.1, x, u, v are variables corresponding to the assignments of entities, relations and implicit relations. t is a vector of binary variables corresponding to the assignments of attributes. a takes values from a pre-defined set of attribute classes if the entity label $z \neq O$, and a equals $\varnothing$ if $z = O$. g_{ia} is a confidence score for the attribute assignment which can be output by a maximum entropy classifier as described in Section 4.3.2. The global consistency constraints include the five types of constraints described in Section 3.2.3 (they can be applied to general entities and relations) and an attribute constraint that enforces the consistency of the attribute assignments: $\sum_a t_{ia} = x_{iz}$ if $z \neq O$. It is also possible to add entity coreference into the framework by treating coreference as a special relation and adding constraints enforcing that only text spans with the same entity type can be coreferential.

Incorporating world knowledge. A common theme in the proposed approaches is leveraging multiple sources of information to make well-informed decisions. A lot of the information we considered comes from linguistic resources, but little comes from existing knowledge about the world. There has been growing interest in utilizing knowledge resources like Wikipedia to help in solving natural language understanding tasks. For example, Wikipedia entities have been shown to be useful for entity coreference resolution (Ratinov and Roth, 2012) and text classification (Vitale et al., 2012). It would be interesting to explore how to use knowledge about real-world entities to help extract opinions, events and other types of information from text. Commonsense knowledge is another important source of knowledge. Recent work (Angeli et al., 2015) has shown the use of certain types of common logical inferences in natural language (e.g., "Heinz Fischer of Austria visits China" entails that "Heinz Fischer visits China") for improving the task of open-domain relation extrac-

tion. To take a step further, we would like to explore the use of richer types of logical inference to help extract a broader range of information.

End-to-end evaluation and user feedback. The ultimate goal of building an information extraction system is to help users better navigate and understand large amounts of text. To achieve this goal, we need to validate the system in end-to-end applications and incorporate user feedback into system development. Therefore, we plan to employ the proposed opinion and event extraction algorithms in a question answering system that aims to answer opinion- or event-oriented questions created by users. We would like to evaluate the effect of using the extracted structured information versus using the raw sentences in answering questions. Furthermore, we want to collect user feedback on the quality of the answers and utilize it to help training information extraction models.

BIBLIOGRAPHY

[Ahn2006] David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8. Association for Computational Linguistics.

[Andrew2006] Galen Andrew. 2006. A hybrid markov/semi-markov conditional random field for sequence segmentation. In *EMNLP*, pages 465–472. Association for Computational Linguistics.

[Angeli et al.2015] Gabor Angeli, Melvin Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *ACL*.

[Bagga and Baldwin1998] Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–6. Citeseer.

[Bejan and Harabagiu2010] Cosmin Adrian Bejan and Sanda Harabagiu. 2010. Unsupervised event coreference resolution with rich linguistic features. In *ACL*, pages 1412–1422. Association for Computational Linguistics.

[Bejan and Harabagiu2014] Cosmin Adrian Bejan and Sanda Harabagiu. 2014. Unsupervised event coreference resolution. *Computational Linguistics*, 40(2):311–347.

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

[Berant et al.2014] Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Brad Huang, Christopher D Manning, Abby Vander Linden, Brittany Harding, and Peter Clark. 2014. Modeling biological processes for reading comprehension. In *Proc. EMNLP*.

[Bethard et al.2004] Steven Bethard, Hong Yu, Ashley Thornton, Vasileios Hatzivassiloglou, and Dan Jurafsky. 2004. Automatic extraction of opinion propositions and their holders. In *2004 AAAI Spring Symposium on Exploring Attitude and Affect in Text*, volume 2224.

[Bhattacharya and Getoor2006] Indrajit Bhattacharya and Lise Getoor. 2006. A latent dirichlet model for unsupervised entity resolution. In *SDM*, volume 5, page 59. SIAM.

[Blei and Frazier2011] David M Blei and Peter I Frazier. 2011. Distance dependent chinese restaurant processes. *The Journal of Machine Learning Research*, 12:2461–2488.

[Blitzer et al.2007] John Blitzer, Mark Dredze, Fernando Pereira, et al. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, volume 7, pages 440–447.

[Breck et al.2007a] E. Breck, Y. Choi, and C. Cardie. 2007a. Identifying expressions of opinion in context. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 2683–2688. Morgan Kaufmann Publishers Inc.

[Breck et al.2007b] Eric Breck, Yejin Choi, and Claire Cardie. 2007b. Identifying expressions of opinion in context. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*.

[Cardie et al.1999] Claire Cardie, Kiri Wagstaff, et al. 1999. Noun phrase coreference as clustering. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 82–89.

[Cardie et al.2003] Claire Cardie, Janyce Wiebe, Theresa Wilson, and Diane J Litman. 2003. Combining low-level and summary representations of opinions for multi-perspective question answering. In *New directions in question answering*, pages 20–27.

[Cardie1997] Claire Cardie. 1997. Empirical methods in information extraction. *AI magazine*, 18(4):65.

[Chang and Collins2011] Yin-Wen Chang and Michael Collins. 2011. Exact decoding of phrase-based translation models through lagrangian relaxation. In *EMNLP*, pages 26–37. Association for Computational Linguistics.

[Chen and Ji2009] Zheng Chen and Heng Ji. 2009. Language specific issue and feature exploration in chinese event extraction. In *NAACL*, pages 209–212. Association for Computational Linguistics.

[Chen et al.2009] Zheng Chen, Heng Ji, and Robert Haralick. 2009. A pairwise event coreference model, feature impact and evaluation for event coreference resolution. In *Proceedings of the Workshop on Events in Emerging Text Types*, pages 17–22. Association for Computational Linguistics.

[Choi and Cardie2008] Yejin Choi and Claire Cardie. 2008. Learning with compositional semantics as structural inference for subsentential sentiment analysis. In *EMNLP*, pages 793–801. Association for Computational Linguistics.

[Choi and Cardie2009] Yejin Choi and Claire Cardie. 2009. Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *EMNLP*, pages 590–598. Association for Computational Linguistics.

[Choi and Cardie2010] Yejin Choi and Claire Cardie. 2010. Hierarchical sequential learning for extracting opinions and their attributes. In *ACL*.

[Choi et al.2005] Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 355–362. Association for Computational Linguistics.

[Choi et al.2006] Y. Choi, E. Breck, and C. Cardie. 2006. Joint extraction of entities and relations for opinion recognition. In *EMNLP*, pages 431–439. Association for Computational Linguistics.

[Cowie and Lehnert1996] Jim Cowie and Wendy Lehnert. 1996. Information extraction. *Communications of the ACM*, 39(1):80–91.

[Crammer et al.2006] Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *The Journal of Machine Learning Research*, 7:551–585.

[Cybulska and Vossen2014a] Agata Cybulska and Piek Vossen. 2014a. Guidelines for ecb+ annotation of events and their coreference. Technical report, Technical report, Technical Report NWR-2014-1, VU University Amsterdam.

[Cybulska and Vossen2014b] Agata Cybulska and Piek Vossen. 2014b. Using a sledgehammer to crack a nut? lexical diversity and event coreference resolution. In *Proceedings of the 9th Language Resources and Evaluation Conference (LREC2014)*, pages 26–31.

[Das et al.2012] Dipanjan Das, André FT Martins, and Noah A Smith. 2012. An exact dual decomposition algorithm for shallow semantic parsing with constraints. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 209–217. Association for Computational Linguistics.

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

[Duchi et al.2010] John Duchi, Elad Hazan, and Yoram Singer. 2010. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159.

[Durrett and Klein2013] Greg Durrett and Dan Klein. 2013. Easy victories and uphill battles in coreference resolution. In *EMNLP*, pages 1971–1982.

[Fan et al.2008] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

[Finkel et al.2006] Jenny Rose Finkel, Christopher D Manning, and Andrew Y Ng. 2006. Solving the problem of cascading errors: Approximate bayesian inference for linguistic annotation pipelines. In *EMNLP*.

[Ganchev and Das2013] Kuzman Ganchev and Dipanjan Das. 2013. Cross-lingual discriminative learning of sequence models with posterior regularization. In *EMNLP*.

[Ganchev et al.2009] Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proceedings of the ACL-IJCNLP*, pages 369–377.

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

[Geman and Geman1984] Stuart Geman and Donald Geman. 1984. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741.

[Ghosh et al.2011] Soumya Ghosh, Andrei B Ungureanu, Erik B Sudderth, and David M Blei. 2011. Spatial distance dependent chinese restaurant processes for image segmentation. In *Advances in Neural Information Processing Systems*, pages 1476–1484.

[Ghosh et al.2014] Soumya Ghosh, Michalis Raptis, Leonid Sigal, and Erik B Sudderth. 2014. Nonparametric clustering with distance dependent hierarchies. In *UAI*.

[Gimpel and Smith2010] Kevin Gimpel and Noah A Smith. 2010. Softmax-margin crfs: Training log-linear models with cost functions. In *NAACL*.

[Goldberg and Zhu2006] Andrew B Goldberg and Xiaojin Zhu. 2006. Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pages 45–52. Association for Computational Linguistics.

[Grishman and Sundheim1996] Ralph Grishman and Beth Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING*, volume 96, pages 466–471.

[Grishman2011] Ralph Grishman. 2011. Information extraction: Capabilities and challenges.

[Haghighi and Klein2007] Aria Haghighi and Dan Klein. 2007. Unsupervised coreference resolution in a nonparametric bayesian model. In *ACL*, volume 45, page 848.

[Haghighi and Klein2010] Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *NAACL*, pages 385–393. Association for Computational Linguistics.

[Hatzivassiloglou and McKeown1997] Vasileios Hatzivassiloglou and Kathleen R McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics*, pages 174–181. Association for Computational Linguistics.

[Hu and Liu2004a] M. Hu and B. Liu. 2004a. Mining opinion features in customer reviews. In *Proceedings of the National Conference on Artificial Intelligence*, pages 755–760. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

[Hu and Liu2004b] Minqing Hu and Bing Liu. 2004b. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

[Humphreys et al.1997] Kevin Humphreys, Robert Gaizauskas, and Saliha Azzam. 1997. Event coreference for information extraction. In *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 75–81. Association for Computational Linguistics.

[Ikeda et al.2010] Daisuke Ikeda, Hiroya Takamura, and Manabu Okumura. 2010. Learning to shift the polarity of words for sentiment classification. *Transactions of the Japanese Society for Artificial Intelligence*, 25(1):50–57.

[Ji and Grishman2008] Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *ACL*, pages 254–262.

[Johansson and Moschitti2010a] Richard Johansson and Alessandro Moschitti. 2010a. Reranking models in fine-grained opinion analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 519–527. Association for Computational Linguistics.

[Johansson and Moschitti2010b] Richard Johansson and Alessandro Moschitti. 2010b. Syntactic and semantic structure for opinion expression detection. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 67–76. Association for Computational Linguistics.

[Johansson and Moschitti2011] Richard Johansson and Alessandro Moschitti. 2011. Extracting opinion expressions and their polarities: exploration of pipelines and joint models. In *Proceedings of the Association for Computational Linguistics: Human Language Technologies: short papers*.

[Johansson and Moschitti2013a] Richard Johansson and Alessandro Moschitti. 2013a. Relational features in fine-grained opinion analysis. *Computational Linguistics*, 39(3):473–509.

[Johansson and Moschitti2013b] Richard Johansson and Alessandro Moschitti. 2013b. Relational features in fine-grained opinion analysis. *Computational Linguistics*, 39(3):473–509.

[Jurafsky et al.2000] Dan Jurafsky, James H Martin, Andrew Kehler, Keith Vander Linden, and Nigel Ward. 2000. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*, volume 2. MIT Press.

[Kanayama and Nasukawa2006] Hiroshi Kanayama and Tetsuya Nasukawa. 2006. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *EMNLP*, pages 355–363. Association for Computational Linguistics.

[Kehler and Kehler2002] Andrew Kehler and Andrew Kehler. 2002. *Coherence, reference, and the theory of grammar*. CSLI publications Stanford, CA.

[Kim and Hovy2004] Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1367. Association for Computational Linguistics.

[Kim and Hovy2006] Soo-Min Kim and Eduard Hovy. 2006. Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of the Workshop on Sentiment and Subjectivity in Text*, pages 1–8. Association for Computational Linguistics.

[Kim and Oh2011] Dongwoo Kim and Alice Oh. 2011. Accounting for data dependencies within a hierarchical dirichlet process mixture model. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 873–878. ACM.

[Kobayashi et al.2007] N. Kobayashi, K. Inui, and Y. Matsumoto. 2007. Extracting aspect-evaluation and aspect-of relations in opinion mining. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1065–1074.

[Lafferty et al.2001] John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*.

[Lazaridou et al.2013] Angeliki Lazaridou, Ivan Titov, and Caroline Sporleder. 2013. A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations. In *ACL (1)*, pages 1630–1639.

[Lee et al.2011] Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 28–34. Association for Computational Linguistics.

[Lee et al.2012] Heeyoung Lee, Marta Recasens, Angel Chang, Mihai Surdeanu, and Dan Jurafsky. 2012. Joint entity and event coreference resolution across

documents. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 489–500. Association for Computational Linguistics.

[Lee et al.2013] Heeyoung Lee, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.

[Liu and Nocedal1989] Dong C Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528.

[Liu et al.2012] K. Liu, L. Xu, and J. Zhao. 2012. Opinion target extraction using word-based translation model. In *EMNLP*. Association for Computational Linguistics.

[Liu et al.2014] Zhengzhong Liu, Jun Araki, Eduard Hovy, and Teruko Mitamura. 2014. Supervised within-document event coreference using information propagation. In *Proceedings of the International Conference on Language Resources and Evaluation*.

[Liu2012] Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

[Luo2005] Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *EMNLP*, pages 25–32.

[Mann and Thompson1988] William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

[Mao and Lebanon2007] Yi Mao and Guy Lebanon. 2007. Isotonic conditional random fields and local sentiment flow. *Advances in neural information processing systems*, 19:961.

[Martins et al.2009] André FT Martins, Noah A Smith, and Eric P Xing. 2009. Concise integer linear programming formulations for dependency parsing. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 342–350. Association for Computational Linguistics.

[Martins et al.2011] André FT Martins, Noah A Smith, Pedro MQ Aguiar, and Mário AT Figueiredo. 2011. Dual decomposition with many overlapping components. In *EMNLP*, pages 238–249. Association for Computational Linguistics.

[McCallum2005] Andrew McCallum. 2005. Information extraction: Distilling structured data from unstructured text. *Queue*, 3(9):48–57.

[McDonald et al.2007] Ryan McDonald, Kerry Hannan, Tyler Neylon, Mike Wells, and Jeff Reynar. 2007. Structured models for fine-to-coarse sentiment analysis. In *ACL*, volume 45, page 432.

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

[Mikolov et al.2013b] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*.

[Munson et al.2005] Art Munson, Claire Cardie, and Rich Caruana. 2005. Optimizing to arbitrary nlp metrics using ensemble selection. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 539–546. Association for Computational Linguistics.

[Nakagawa et al.2010] Tetsuji Nakagawa, Kentaro Inui, and Sadao Kurohashi. 2010. Dependency tree-based sentiment classification using crfs with hidden variables. In *NAACL*, pages 786–794. Association for Computational Linguistics.

[Okanohara et al.2006] Daisuke Okanohara, Yusuke Miyao, Yoshimasa Tsuruoka, and Jun'ichi Tsujii. 2006. Improving the scalability of semi-markov conditional random fields for named entity recognition. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 465–472. Association for Computational Linguistics.

[Pang and Lee2004] Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *ACL*, page 271. Association for Computational Linguistics.

[Pang and Lee2005] Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL*, pages 115–124. Association for Computational Linguistics.

[Pang and Lee2008] Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

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

[Polanyi and Zaenen2006] Livia Polanyi and Annie Zaenen. 2006. Contextual valence shifters. In *Computing attitude and affect in text: Theory and applications*, pages 1–10. Springer.

[Poon and Domingos2007] Hoifung Poon and Pedro Domingos. 2007. Joint inference in information extraction. In *AAAI*, volume 7, pages 913–918.

[Poon and Vanderwende2010] Hoifung Poon and Lucy Vanderwende. 2010. Joint inference for knowledge extraction from biomedical literature. In *NAACL*, pages 813–821. Association for Computational Linguistics.

[Popescu and Etzioni2005] A.M. Popescu and O. Etzioni. 2005. Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 339–346. Association for Computational Linguistics.

[Pradhan et al.2014] Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *ACL*, pages 22–27.

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

[Punyakanok et al.2004] V. Punyakanok, D. Roth, W. Yih, and D. Zimak. 2004. Semantic role labeling via integer linear programming inference. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1346. Association for Computational Linguistics.

[Punyakanok et al.2008] V. Punyakanok, D. Roth, and W. Yih. 2008. The impor-

tance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2):257–287.

[Qiu et al.2009] Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2009. Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st international jont conference on Artifical intelligence*, pages 1199–1204. Morgan Kaufmann Publishers Inc.

[Qiu et al.2011] G. Qiu, B. Liu, J. Bu, and C. Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational linguistics*, 37(1):9–27.

[Qu et al.2012] Lizhen Qu, Rainer Gemulla, and Gerhard Weikum. 2012. A weakly supervised model for sentence-level semantic orientation analysis with multiple experts. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 149–159. Association for Computational Linguistics.

[Raghunathan et al.2010] Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *EMNLP*, pages 492–501. Association for Computational Linguistics.

[Rahman and Ng2011] Altaf Rahman and Vincent Ng. 2011. Coreference resolution with world knowledge. In *ACL*, pages 814–824. Association for Computational Linguistics.

[Ratinov and Roth2012] Lev Ratinov and Dan Roth. 2012. Learning-based multi-sieve co-reference resolution with knowledge. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1234–1244. Association for Computational Linguistics.

[Reichart and Barzilay2012] Roi Reichart and Regina Barzilay. 2012. Multi event extraction guided by global constraints. In *NAACL*, pages 70–79. Association for Computational Linguistics.

[Riedel and Clarke2006] Sebastian Riedel and James Clarke. 2006. Incremental integer linear programming for non-projective dependency parsing. In *EMNLP*, pages 129–137. Association for Computational Linguistics.

[Riloff et al.2003] Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learn-

ing subjective nouns using extraction pattern bootstrapping. In *Proceedings of Natural language learning at HLT-NAACL*.

[Ritter et al.2012] Alan Ritter, Oren Etzioni, Sam Clark, et al. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM.

[Roth and Frank2012] Michael Roth and Anette Frank. 2012. Aligning predicate argument structures in monolingual comparable texts: A new corpus for a new task. In *SemEval*, pages 218–227. Association for Computational Linguistics.

[Roth and Yih2004] D. Roth and W. Yih. 2004. *A linear programming formulation for global inference in natural language tasks*. Defense Technical Information Center.

[Roth and Yih2007] Dan Roth and Wen-tau Yih. 2007. Global inference for entity and relation identification via a linear programming formulation. *Introduction to statistical relational learning*, pages 553–580.

[Ruppenhofer et al.2008] J. Ruppenhofer, S. Somasundaran, and J. Wiebe. 2008. Finding the sources and targets of subjective expressions. In *Proceedings of LREC*.

[Rush et al.2010] Alexander M Rush, David Sontag, Michael Collins, and Tommi Jaakkola. 2010. On dual decomposition and linear programming relaxations for natural language processing. In *EMNLP*.

[Sarawagi and Cohen2004] Sunita Sarawagi and William W Cohen. 2004. Semimarkov conditional random fields for information extraction. *Advances in Neural Information Processing Systems*, 17:1185–1192.

[Singh et al.2010] Sameer Singh, Michael Wick, and Andrew McCallum. 2010. Distantly labeling data for large scale cross-document coreference. *arXiv preprint arXiv:1005.4298*.

[Singh et al.2013] Sameer Singh, Sebastian Riedel, Brian Martin, Jiaping Zheng, and Andrew McCallum. 2013. Joint inference of entities, relations, and coreference. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 1–6. ACM.

[Sirts et al.2014] Kairit Sirts, Jacob Eisenstein, Micha Elsner, and Sharon Goldwater. 2014. Pos induction with distributional and morphological information using a distance-dependent chinese restaurant process. In *ACL*.

[Snyder and Barzilay2007] Benjamin Snyder and Regina Barzilay. 2007. Multiple aspect ranking using the good grief algorithm. In *HLT-NAACL*, pages 300–307.

[Socher et al.2011a] Richard Socher, Andrew L Maas, and Christopher D Manning. 2011a. Spectral chinese restaurant processes: Nonparametric clustering based on similarities. In *International Conference on Artificial Intelligence and Statistics*, pages 698–706.

[Socher et al.2011b] Richard Socher, Jeffrey Pennington, Eric H Huang, Andrew Y Ng, and Christopher D Manning. 2011b. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *EMNLP*, pages 151–161. Association for Computational Linguistics.

[Socher et al.2013] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP*.

[Somasundaran et al.2008] Swapna Somasundaran, Janyce Wiebe, and Josef Ruppenhofer. 2008. Discourse level opinion interpretation. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 801–808. Association for Computational Linguistics.

[Somasundaran et al.2009] Swapna Somasundaran, Galileo Namata, Janyce Wiebe, and Lise Getoor. 2009. Supervised and unsupervised methods in employing discourse relations for improving opinion polarity classification. In *EMNLP*, pages 170–179. Association for Computational Linguistics.

[Srikumar and Roth2011] Vivek Srikumar and Dan Roth. 2011. A joint model for extended semantic role labeling. In *EMNLP*, pages 129–139. Association for Computational Linguistics.

[Stoyanov and Cardie2006] Veselin Stoyanov and Claire Cardie. 2006. Partially supervised coreference resolution for opinion summarization through structured rule learning. In *EMNLP*, pages 336–344. Association for Computational Linguistics.

[Stoyanov and Cardie2008] Veselin Stoyanov and Claire Cardie. 2008. Topic identification for fine-grained opinion analysis. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 817–824. Association for Computational Linguistics.

[Stoyanov et al.2009] Veselin Stoyanov, Nathan Gilbert, Claire Cardie, and Ellen Riloff. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 656–664. Association for Computational Linguistics.

[Surdeanu et al.2007] Mihai Surdeanu, Lluís Màrquez, Xavier Carreras, and Pere R Comas. 2007. Combination strategies for semantic role labeling. *Journal of Artificial Intelligence Research*, pages 105–151.

[Taboada and Mann2006] Maite Taboada and William C Mann. 2006. Rhetorical structure theory: Looking back and moving ahead. *Discourse studies*, 8(3):423–459.

[Taboada et al.2008] Maite Taboada, Kimberly Voll, and Julian Brooke. 2008. Extracting sentiment as a function of discourse structure and topicality. *Simon Fraser Univeristy School of Computing Science Technical Report*.

[Täckström and McDonald2011a] Oscar Täckström and Ryan McDonald. 2011a. Discovering fine-grained sentiment with latent variable structured prediction models. In *Advances in Information Retrieval*, pages 368–374. Springer.

[Täckström and McDonald2011b] Oscar Täckström and Ryan McDonald. 2011b. Semi-supervised latent variable models for sentence-level sentiment analysis. In *ACL*, pages 569–574. Association for Computational Linguistics.

[Teh et al.2006] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476).

[Trivedi and Eisenstein2013] Rakshit Trivedi and Jacob Eisenstein. 2013. Discourse connectors for latent subjectivity in sentiment analysis. In *Proceedings of NAACL-HLT*, pages 808–813.

[Tsochantaridis et al.2004] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten

Joachims, and Yasemin Altun. 2004. Support vector machine learning for interdependent and structured output spaces. In *ICML*, page 104. ACM.

[Turney2002] Peter D Turney. 2002. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *ACL*, pages 417–424. Association for Computational Linguistics.

[Vilain et al.1995] Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52.

[Vitale et al.2012] Daniele Vitale, Paolo Ferragina, and Ugo Scaiella. 2012. Classification of short texts by deploying topical annotations. In *Advances in Information Retrieval*, pages 376–387. Springer.

[Wang and Manning2013] Sida Wang and Christopher Manning. 2013. Fast dropout training. In *ICML*, pages 118–126.

[Wick et al.2012] Michael Wick, Sameer Singh, and Andrew McCallum. 2012. A discriminative hierarchical model for fast coreference at large scale. In *ACL*, pages 379–388. Association for Computational Linguistics.

[Wiebe et al.1999] Janyce M Wiebe, Rebecca F Bruce, and Thomas P O'Hara. 1999. Development and use of a gold-standard data set for subjectivity classifications. In *ACL*, pages 246–253. Association for Computational Linguistics.

[Wiebe et al.2001] Janyce Wiebe, Theresa Wilson, and Matthew Bell. 2001. Identifying collocations for recognizing opinions. In *Proceedings of the ACL-01 Workshop on Collocation: Computational Extraction, Analysis, and Exploitation*, pages 24–31.

[Wiebe et al.2005] J. Wiebe, T. Wilson, and C. Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2):165–210.

[Wiebe1994] Janyce M Wiebe. 1994. Tracking point of view in narrative. *Computational Linguistics*, 20(2):233–287.

[Wilson et al.2005a] Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005a. Opinionfinder: A system for subjectivity analy-

sis. In *Proceedings of HLT/EMNLP on interactive demonstrations*, pages 34–35. Association for Computational Linguistics.

[Wilson et al.2005b] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005b. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354. Association for Computational Linguistics.

[Wilson et al.2005c] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005c. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*. Vancouver, Canada.

[Wilson et al.2009] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2009. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics*, 35(3):399–433.

[Wilson2008a] Theresa Wilson. 2008a. *Fine-Grained Subjectivity Analysis*. Ph.D. thesis, Ph. D. thesis, University of Pittsburgh. Intelligent Systems Program.

[Wilson2008b] Theresa Ann Wilson. 2008b. *Fine-grained subjectivity and sentiment analysis: recognizing the intensity, polarity, and attitudes of private states*. ProQuest.

[Wolfe et al.2015] Travis Wolfe, Mark Dredze, and Benjamin Van Durme. 2015. Predicate argument alignment using a global coherence model. In *NAACL*.

[Wu et al.2009] Y. Wu, Q. Zhang, X. Huang, and L. Wu. 2009. Phrase dependency parsing for opinion mining. In *EMNLP*, pages 1533–1541. Association for Computational Linguistics.

[Yang and Cardie2012] B. Yang and C. Cardie. 2012. Extracting opinion expressions with semi-markov conditional random fields. In *EMNLP*. Association for Computational Linguistics.

[Yang and Cardie2013] Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *ACL (1)*, pages 1640–1649.

[Yang and Cardie2014a] Bishan Yang and Claire Cardie. 2014a. Context-aware

learning for sentence-level sentiment analysis with posterior regularization. In *ACL*, pages 325–335.

[Yang and Cardie2014b] Bishan Yang and Claire Cardie. 2014b. Joint modeling of opinion expression extraction and attribute classification. *Transactions of the Association for Computational Linguistics*, 2:505–516.

[Yessenalina and Cardie2011] Ainur Yessenalina and Claire Cardie. 2011. Compositional matrix-space models for sentiment analysis. In *EMNLP*.

[Yu and Hatzivassiloglou2003] Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *EMNLP*, pages 129–136. Association for Computational Linguistics.

[Zhang et al.2010] Lei Zhang, Bing Liu, Suk Hwan Lim, and Eamonn O'Brien-Strain. 2010. Extracting and ranking product features in opinion documents. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1462–1470. Association for Computational Linguistics.

[Zhang et al.2013] Qi Zhang, Jin Qian, Huan Chen, Jihua Kang, and Xuanjing Huang. 2013. Discourse level explanatory relation extraction from product reviews using first-order logic. In *EMNLP*.

[Zhao et al.2008] Jun Zhao, Kang Liu, and Gen Wang. 2008. Adding redundant features for crfs-based sentence sentiment classification. In *EMNLP*.

[Zhou et al.2011] Lanjun Zhou, Binyang Li, Wei Gao, Zhongyu Wei, and Kam-Fai Wong. 2011. Unsupervised discovery of discourse relations for eliminating intra-sentence polarity ambiguities. In *EMNLP*, pages 162–171. Association for Computational Linguistics.