SEMI-SUPERVISED LEARNING FOR IDENTIFYING OPINIONS IN WEB CONTENT

Ning Yu

Submitted to the faculty of the University Graduate School

in partial fulfillment of the requirements

for the degree

Doctor of Philosophy

in the School of Library and Information Science,

Indiana University

April 2011

Accepted by the Graduate Faculty, Indiana University, in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Doctoral Committee

_____

Elin K. Jacob, Ph.D.

_____

Kiduk Yang, Ph.D.

_____

Sandra Kübler, Ph.D.

_____

Ying Ding, Ph.D.

March 4, 2011

Acknowledgments

This dissertation would not have been possible without the support of many people, and I appreciate their contributions to the completion of this dissertation.

I am thankful to all my dissertation committee members for being my sounding board and guiding me in becoming a qualified researcher.

I am indebted to my dissertation advisor, Dr. Elin K. Jacob, who offered me precious guidance and tremendous support through my doctoral study. She always found time to provide me with detailed feedback and suggestions to make this thesis a fine piece of work. She also provided generous financial support for several of my conference trips related to the dissertation research.

I am heartily thankful to my Ph.D. minor advisor, Dr. Sandra Kübler, for her critical reading of the dissertation and her constructive comments. She sparked my interest in co-training, an important methodology used in the dissertation research, and offered support far beyond her responsibility as a minor advisor.

I am grateful to my former Ph.D. advisor, Dr. Kiduk Yang, for introducing me to the research in information retrieval and for supervising me on various project that laid the foundation for my dissertation work. Without his encouragement, supervision and support, I could not have gone through the most difficult time in my Ph.D. years and accomplished the dissertation.

I would like to thank Dr. Ying Ding for offering a different perspective on the research problem.

I would also like to thank Dr. Dagobert Soergel, Dr. Werner Ceusters and Dr. William J. Rapaport of the University of Buffalo for their useful feedback on the preliminary results of the dissertation research.

I am grateful to my colleagues at the School of Library and Information Science for providing a friendly and supportive research environment. Special thanks go to Ruby Huang, who provided me with necessary prodding.

My sincere thanks go to Bob Carpenter and Peter Reutemann for providing detailed answers to my questions regarding the use of Weka and Lingpipe. Special thanks go to Ching-Hao Mao at the National Taiwan University of Science and Technology for discussing co-training implementation in Weka and for sharing his java code as a reference. I would like to thank Yuyin Sun for providing the java code used for calling character-based language model in Lingpipe and running POS tagger against movie review and news datasets. While I am grateful for all the assistance I received, the responsibility for any errors in system implementation must lie with me alone.

I cannot exaggerate my appreciation for the outstanding support from my family and friends. They were there for me from the start to the end with great patience and unconditional understanding.

Last but not least, I want to thank our local Asian grocery store on 10th Street for carrying Chinese sweets, which cheered me and kept me writing.

Ning Yu

SEMI-SUPERVISED LEARNING
FOR IDENTIFYING OPINIONS IN WEB CONTENT

Opinions published on the World Wide Web (Web) offer opportunities for detecting personal attitudes regarding topics, products, and services. The opinion detection literature indicates that both a large body of opinions and a wide variety of opinion features are essential for capturing subtle opinion information. Although a large amount of opinion-labeled data is preferable for opinion detection systems, opinion-labeled data is often limited, especially at sub-document levels, and manual annotation is tedious, expensive and error-prone. This shortage of opinion-labeled data is less challenging in some domains (e.g., movie reviews) than in others (e.g., blog posts). While a simple method for improving accuracy in challenging domains is to borrow opinion-labeled data from a non-target data domain, this approach often fails because of the domain transfer problem: Opinion detection strategies designed for one data domain generally do not perform well in another domain. However, while it is difficult to obtain opinion-labeled data, unlabeled user-generated opinion data are readily available. Semi-supervised learning (SSL) requires only limited labeled data to automatically label unlabeled data and has achieved promising results in various natural language processing (NLP) tasks, including traditional topic classification; but SSL has been applied in only a few opinion detection studies. This study

investigates application of four different SSL algorithms in three types of Web content: edited news articles, semi-structured movie reviews, and the informal and unstructured content of the blogosphere. SSL algorithms are also evaluated for their effectiveness in sparse data situations and domain adaptation. Research findings suggest that, when there is limited labeled data, SSL is a promising approach for opinion detection in Web content. Although the contributions of SSL varied across data domains, significant improvement was demonstrated for the most challenging data domain—the blogosphere—when a domain transfer-based SSL strategy was implemented.

Table of Contents

List of Tables

List of Figures

1 Introduction

The rapid growth of freely accessible and easily customizable applications on the World Wide Web (Web) has made it easy and fun for people to share their experiences, knowledge and opinions[1]. Retail websites such as Amazon.com and review aggregators such as Yelp.com collect customer reviews on specific products or services while blogs and social networking sites such as Twitter and Facebook allow users to publish opinions on an infinite array of topics ranging from the benefits of blueberries to the U.S. presidential election. According to a 2008 report on the state of the blogosphere published by Technorati (Sifry, 2008), more than 80% of bloggers post brand or product reviews, with 37% doing so frequently, and more than 60% of bloggers read other blogs to learn about products or services.

Researchers from different communities have been working in the area of opinion mining since the late 1990s. Dave, Lawrence and Pennock (2003), who coined the phrase "opinion mining," described a tool for mining online product reviews that was intended to automate the sequence of processing "a set of search results for a given item, generating a list of product attributes (quality, features, etc.) and aggregating opinions about each of them (poor, mixed, good)" (p. 519). This tool exemplified one possible opinion mining application and demonstrated a general procedure for opinion mining.

[1] Web content that is generated by users is called user-generated content or consumer-generated media (Liu, 2007).

To date, a number of opinion mining tasks have been explored, including differentiating opinions from facts (Wiebe, Wilson, Bruce, Bell, & Martin, 2004; Wilson, Pierce, & Wiebe, 2003; Yang, Yu, & Zhang, 2007; Zhang, Yu, & Meng, 2007); detecting positive and negative opinion polarity (Abbasi, Chen, & Salem, 2008; Cui, Mittal, & Datar, 2006; Kim & Hovy, 2004; Koppel & Shtrimberg, 2006; Ku & Chen, 2007; Liu, 2007; Pang, Lee, & Vaithyanathan, 2002); determining opinion strength (Tsou, Yuen, Kwong, Lai, & Wong, 2005; Wilson, Wiebe, & Hwa, 2004); and identifying other opinion properties (Bethard, Yu, Thornton, Hatzivassiloglou, & Jurafsky, 2006; Kim & Hovy, 2006; Ku & Chen, 2007; Li, Bontcheva, & Cunningham, 2007). For all of these tasks, opinion detection is fundamental, especially in data domains such as the blogosphere, where each blog is frequently a mixture of facts and opinions.

1.1 *Opinion* and Related Terms

The Merriam-Webster online dictionary (2008) defines *opinion* as "a view, judgment, or appraisal formed in the mind about a particular matter." This general definition shows that opinions are subjective and that they are always about something. In the context of opinion mining, however, there is no widely accepted definition of *opinion* beyond the general agreement that an opinion is something that is not fact. Researchers typically attempt to understand *opinion* by decomposing it into a set of components and attributes. For example, Liu (2007) listed three main components of an opinion: the opinion holder, the object of the opinion, and the opinion itself. *Opinion holder* refers to the person or organization

that holds an opinion. In the case of user-generated content, opinion holders are, by default, the authors of the original documents and any follow-up comments, and they are not of great interest to researchers. The *objec*t is an entity about which the opinion is expressed, and it is usually identified with a search term by Internet users. An object may contain several sub-components or attributes: For example, when someone writes a review for a digital camera (the object), he could write about optical zoom, battery life, or any other specific aspect of the camera. Within Liu's framework, the *opinion* is a textual commentary about an entity offered by a person or organization.

Opinion is sometimes used interchangeably with sentiment and subjectivity. *Sentiment* is the preferred term in the natural language processing (NLP) community, where sentiment analysis tends to refer to the task of polarity identification (i.e., the task of identifying positive, negative or mutual opinions). The *subjectivity* of an expression refers to the degree to which it reflects an individual's personal opinions, evaluations, emotions, or speculations. Nonetheless, *sentiment* and *subjectivity* are used interchangeably in a series of seminal studies (Riloff & Wiebe, 2003; Wiebe, Bruce, & O'Hara, 1999; Wiebe & Mihalcea, 2006; Wiebe & Wilson, 2002; Wiebe et al., 2004; Wiebe, Wilson, & Cardie, 2005). For example, in a recent survey of opinion mining and sentiment analysis by Pang and Lee (2008), opinion mining is defined as the "computational treatment of opinion, sentiment, and subjectivity in text" (p. 8), highlighting the interchangeability of these terms in the context of opinion mining.

Related terms such as *affect*, *attitude*, *emotion* and *mood* are also encountered in the literature (Grefenstette, Qu, Shanahan, & Evans, 2004; Hancock, Landrigan, & Silver, 2007; Holzman & Pottenger, 2003; Liu, Lieberman, & Selker, 2003; Mishne, 2005; Tokuhisa & Terashima, 2006; Wiebe et al., 2005; Zhang, Barnden, Hendley, & Wallington, 2006). These terms are similar in that they are related to the more general term *feeling*, but they can frequently be distinguished by the community of use: For example, *emotion* or *affect* is frequently used in the field of cognitive science (Pfeifer, 1988; Singer & Salovey 1988).

1.2 Characteristics of Opinions

In contrast to statements of fact, opinions are expressed in a less straightforward and more diverse manner. Nigam and Hurst (2004) defined a spatial model with four dimensions—explicitness, realness, restriction and attribution—to demonstrate the subtle nature of opinion expressions. Examples of these dimensions include:

1. Explicitness
    - This camera is great! (explicit opinion)
    - I returned this product after a week. (implicit opinion)
    - Go to this restaurant if you like raw food. (irony/sarcasm)
2. Realness
    - I love my new camera. (real-world concept)
    - I am looking for my next wonderful dress. (hypothetical concept)

3. Restriction

   - A family cruise trip is going to be wonderful! (time restricted)
   - I may enjoy the music. (modal restriction)
   - I will like this phone only if it is smaller. (condition restriction)

4. Attribution

   - I think John will love this song. (author's judgment, but John may hate the song!)

Without careful attention to the distinction between opinion and fact, it can sometimes be problematic for humans to make consistent judgments regarding less explicit opinion expressions. This has been confirmed by relatively low rates of inter-annotator agreement for manual annotation of opinion sentences, which range between 70% and 80% (Gamon, Aue, Corston-Oliver, & Ringger, 2005; Tokuhisa & Terashima, 2006; Wiebe et al., 2005). Obviously, separating opinion from fact can be even more difficult for computers.

Wiebe (2000) observed that people become creative when expressing opinions and tend to use uncommon or rare term patterns. In a collection of data from the *Wall Street Journal*, the difference in proportion of unique words occurring in opinions and non-opinions was significant at $p<0.001$ ($z>=22$) (Wiebe et al., 2004). The use of uncommon (i.e., low frequency) terms in expressing opinions is also notable in user-generated content. Yang et al. (2007) captured uncommon and creative word forms used in the blogosphere and categorized them as intentionally misspelled words (e.g., "luv," "hizzarious"),

compound words (e.g., “metacool,” “crazygood”), repeated-character words (e.g., “soooo,” “fantaastic,” “grrrreat”), or combinations of the three (e.g., “metacoool”).

1.3 Opinion Detection

Opinion detection is the task of detecting opinion-bearing documents or opinion-bearing portions of a document, normally based on the presence or absence of opinion indicator(s). Opinion indicators are sometimes called opinion cues, opinion markers, or opinion features. For ease of discussion, *opinion-bearing features* will be used throughout this paper.

In most cases, opinion detection is treated as a problem of binary classification utilizing the categories opinion and fact and is evaluated by classification accuracy. There are also cases, especially in opinion retrieval, when opinion detection involves assigning scores to fragments of text to indicate how likely it is that they are opinions. Differences between these two approaches involve final delivery and evaluation measures, which are determined by the intended application of opinion detection.

An opinion-mining task closely related to opinion detection is *polarity detection*, which identifies the semantic orientation (i.e., positive or negative) of the target text. The techniques developed for opinion detection can be easily adapted to polarity detection and, in some cases, opinion detection subsumes polarity detection when the goal is to identify positive, negative and neutral pieces of information in a given text (Chesley, Vincent, Xu, & Srihari, 2006; Niu, Zhu,

Li, & Hirst, 2005). Although many studies have skipped the step of opinion detection and performed polarity detection directly, Esuli and Sebastiani (2006) suggest that opinion detection is a more fundamental and more difficult task than polarity detection and that polarity detection benefits from prior opinion detection.

Opinion detection can also be understood as a sub-task of other, non-factual information detection tasks. *Genre detection*, which identifies types of documents (Karlgren & Cutting, 1994), subsumes opinion detection when a genre (e.g., editorials) is inherently more subjective than another (e.g., technology reports). *Question answering*, an information retrieval task that returns actual answers rather than whole documents, may also subsume the task of opinion detection when the question is opinion-oriented (e.g., "What do people think about the US-Iraq war?"). *Appraisal extraction*[2] (Bloom, Garg, & Argamon, 2007) and *affective computing*[3] (Picard, 1997), both of which include the task of identifying attitudes and emotions (Bloom et al., 2007; Hancock et al., 2007; Liu et al., 2003; Zhang et al., 2006), are also closely related to opinion detection.

1.4 Four Dimensions of Opinion Detection Systems

There are four basic factors or aspects to consider for each specific opinion detection system: target data domain(s), task, level of granularity, and methods.

[2] The main task of appraisal extraction is to extract text that expresses an attitude towards an object (i.e., appraisal expressions). An appraisal expression typically consists of an appraisal of an object, the target of the appraisal, and the source of the appraisal.

[3] Affective computing is a sub-field in artificial intelligence that attempts to imbue computers with the ability to recognize, to understand, and even to express human emotions. One major area in affective computing is automatic detection and recognition of emotional (i.e., affective) information.

Theoretically, an opinion detection system can be a combination of any values in each of the four dimensions; practically, these dimensions are interactional and the value of one dimension will often affect decisions regarding other dimensions.

1.4.1 Data Domains

Three types of Web content have been explored in opinion detection studies: news articles, online reviews, and online discourse in blogs or discussion forums[4]. These three types of Web content differ from one another in terms of structure, the use of formal or informal language, and the proportion of opinions.

News articles are formal, well-structured documents written by professional journalists. Although objectivity is expected in news reports, they often include editorials and reviews as well as opinions quoted from celebrities and authorities.

Online reviews are normally focused on a predefined target such as a movie or a product and are sometimes organized under sections such as pros, cons and overall rating. These reviews generally use informal language. They are often written in short or even incomplete sentences and are rich in opinions on different aspects of the reviewed products or services.

Blogs, or online discussions, are also written in informal language, but they do not organize opinions in separate sections and are not necessarily focused on one topic. Consequently, there are no categorical or structural cues for locating opinions in blogs.

[4] Occasionally, email messages and instant messaging (IM) conversations are included in this domain.

1.4.2 Tasks

Opinion detection tasks are described as *targeted* (i.e., topical) or *non-targeted* (i.e., non-topical) tasks depending on whether the task involves looking for opinions associated with specific topics. Conceptually, the recognition of an opinion and the determination of its associated topic occur simultaneously, but opinion recognition and topic determination are frequently treated as two separate problems requiring different theories and methods. Unless otherwise specified, opinion detection in this paper refers to non-targeted (i.e., non-topical) opinion detection.

1.4.3 Levels of Granularity

Opinion detection can operate at several levels depending on the granularity of the target text unit: the term level, the expression level, the passage level, and the document level. The level of granularity in opinion detection tasks is usually determined by the purpose of the application. Practically, granularity is also restricted by the availability of a labeled data collection: The finer the level at which the data collection has been labeled, the more levels of granularity on which the opinion detection system can be built.

Term-level opinion detection determines whether a single word or phrase carries opinion information within the sentence or document in which the term appears. In real-world applications, term-level opinion detection is rarely the

ultimate goal. Instead, a set of terms serves as a group of opinion-bearing features for higher-level opinion detection.

Expression-level opinion detection identifies the exact portion of text that is directly responsible for the opinion attribute. For example, in the sentence "*I am happy* for choosing this camera," the opinion expression is indicated by italics. As suggested by Wiebe, Bruce, Bell, Martin, and Wilson (2001), expression-level opinion detection should only be used during the process of manually labeling a data collection since the lack of explicit boundaries for opinion expressions presents challenges for automatic labeling.

Passage-level opinion detection is the most popular of the four levels. A passage can include a single sentence, multiple sentences, or a text unit of any arbitrary length. Although automated passage-level opinion detection is feasible, it is challenging both because of the difficulty of precisely truncating the passage boundary and because of the sparsity of opinion-bearing features in short text units.

Document-level opinion detection is directly related to real-world applications such as retrieving opinion posts from the blogosphere. In blog data, an opinion detector trained on documents is not expected to perform well since the proportion of opinions in a blog post can be very low. Instead, document-level opinion detectors usually deliver document-level predictions by integrating predictions made by term-level, expression-level and passage-level opinion predictors.

Other levels of granularity in opinion detection include multi-document collections and speech, which is defined as a "continuous single-speaker segment of text" (Thomas, Pang, & Lee, 2006, p. 328).

1.4.4 Methods

Current methods employed in opinion detection fall into three major areas — natural language processing, supervised learning, and information retrieval — depending on the sources of evidence on which they rely and how they leverage that evidence.

Natural Language Processing (NLP) methods utilize linguistic knowledge for word relationship identification and syntactic parsing. These methods work well at the sentence level but are computationally expensive.

Supervised Learning (SL) algorithms are well-established machine learning techniques that can automatically learn important opinion-bearing features from a labeled data collection and/or build opinion detection models in terms of feature combinations. Both the scale and quality of a collection of labeled data play an important role in this approach. The ideal training set for SL techniques is a sample of the target data domain that is representative with respect to vocabulary, word distribution, and other characteristics.

Information Retrieval (IR) methods leverage term occurrence statistics (i.e., term frequency) and similarity relationships to retrieve relevant objects. They can be applied in harvesting opinion training data and in creating an opinion lexicon

as well as in opinion scoring. One straightforward IR application is querying Web resources (e.g., Wikipedia.com, Eopinion.com) with certain keywords (e.g., "Skype") in order to extract opinion or non-opinion documents in the target domain. Occasionally, linkage analysis, which utilizes co-citation relationships and link structures, is used to identify the polarity of opinions.

2 Literature Review of Opinion Detection and Semi-Supervised Learning

A review of major studies in opinion detection illustrates the evolution of opinion detection and opinion detection strategies and points to the challenges facing opinion detection and the questions remaining to be addressed in the field. The review of semi-supervised learning (SSL), a methodology central to this research, explores the most common SSL algorithms and their assumptions, benefits, limitations, and major applications as well as related work in the area of opinion detection.

2.1 Evolution of Opinion Detection

The literature of opinion detection bears witness to three inter-related stages in the development of opinion detection: subjectivity analysis, review-based sentiment analysis, and targeted opinion detection. These three stages represent the evolution of opinion detection from a purely linguistic problem to a practical task in the real world.

During the late 1990s, a group of researchers began studying the subjectivity of language (Hatzivassiloglou & Wiebe, 2000; Wiebe, 2000; Wiebe et al., 1999). They understood subjectivity as a pragmatic, sentence-level feature (Hatzivassiloglou & Wiebe, 2000) and focused on studying the general

characteristics of subjective language and testing the feasibility and reliability of automatically identifying various subjectivity cues.

With the increasing volume of online movie and product reviews available on the Web, researchers in opinion detection began to perform review-based sentiment analysis (Gamon, 2004; Nasukawa & Yi, 2003; Pang et al., 2002; Yi, Nasukawa, Bunescu, & Niblack, 2003). Due to the large proportion of opinions in reviews, one assumption often implicit in sentiment analysis was that all sentences in reviews express opinions. The main task at this stage was therefore to classify reviews as favorable (positive) or unfavorable (negative); and the dominant approach shifted from linguistic knowledge acquisition to supervised classification to better deal with large-scale data.

The most recent trend in opinion detection moved from classifying relatively structured review data to considering associations between opinions and their associated topics. Research on targeted opinion detection in the blogosphere has been conducted as part of the Blog track at the Text REtrieval Conference (TREC). TREC's Blog track is an information retrieval contest, held annually since 2006, that requires participants to investigate targeted opinion detection by integrating traditional topical retrieval with opinion retrieval. In the Blog track, participants try out various strategies for finding opinion blog posts on specific topics in a standardized environment using the same partially labeled data collection, the same search topics, and the same evaluation measures.

The evolution of opinion detection can be mapped onto four dimensions: target domain, task, granularity, and method. Table 1 summarizes the dominant values for each dimension in each stage in the evolution of opinion detection. For example, the dominant method used in opinion detection has shifted from the use of natural language processing (NLP) in subjective analysis to a fusion of NLP, supervised learning (SL) and information retrieval (IR) methods in targeted opinion detection.

Table 1

Mapping the Evolution of Opinion Detection onto Four Dimensions

	Stage		
Dimensions	Subjective Analysis	Review-Based Sentiment Analysis	Targeted Opinion Detection
Target Domain	Online news	Online reviews	Online discourse
Task	Non-targeted	Non-targeted *	Targeted
Granularity	Term, Sentence	Sentence, Document	Sentence, Document
Method	NLP	NLP, SL	NLP, IR, SL

*polarity detection dominates this stage.

2.2 Main Approaches in Opinion Detection

The ad hoc rule-based approach, sometimes known as the lexicon-based approach (Ounis, Macdonald, & Soboroff, 2008), and the machine learning-based approach are the two major types of opinion detection strategies that have been used by TREC's Blog track participants in designing blog opinion detection

systems. The opinion detection literature indicates that both of these approaches benefit from the large number and great variety of opinion-bearing features used as evidence in opinion detection.

Opinion evidence can be knowledge-based, statistical/empirical, or style-based. *Knowledge-based* opinion evidence is based on researchers' understandings and observations of opinion expression. The most intuitive and widely used opinion-bearing features in knowledge-based evidence are the words or phrases that are semantically associated with opinions (e.g., "love"). These features are useful for detecting explicit opinions that contain such words, but they fail in the case of opinions that are implicitly expressed, as in the sentence "This car has a high accident ratio." Knowledge-based opinion-bearing features are extracted manually or semi-manually and are normally independent of the target data domain; they are of high quality but limited quantity. Examples of opinion-bearing features extracted as knowledge-based opinion evidence include certain class of verbs (e.g., "agree") that may be used in propositions to initiate an opinion statement; low frequency terms based on the creative and unique nature of opinion language (e.g., "soooo"); subsets of adjectives (e.g., semantic oriented adjectives: "good", "bad"); common sense knowledge (e.g., "riding a roller coaster" often indicates excitement and thus positive opinion); and syntactic patterns based on manually selected word dependency relations (e.g., <relation, headWord, dependentWord>).

Statistical/empirical evidence, often used in information retrieval and topical classification, can also be used for opinion detection. The basic assumption behind use of this evidence is that words, phrases, or patterns occurring more frequently in opinion expressions than in non-opinion expressions are more likely to be good indicators of opinions. In contrast to most knowledge-based opinion-bearing features, statistical opinion-bearing features are not always intuitive (e.g., "try the", "off", "just"), and they are usually automatically extracted from specific data collections. Examples of opinion-bearing features based on statistical evidence include bag-of-words (i.e., a simple collection of words, regardless of word order) and high order n-grams (i.e., a sequence of n>1 adjacent words within a sentence). Although opinion-bearing features based on statistical evidence are simple and straightforward, they can be as valuable as complex features when there is a large amount of labeled data. Based on their experiments, Ng, Dasgupta, and Arifin (2006) concluded that "high order n-grams and dependency-based features capture essentially the same information" (p. 617).

Writing *style* and document *structure*, based on research in linguistics and information retrieval, are other sources of opinion evidence. Some instances of style-based opinion detection are the shortened word forms or emoticons used on the Web (e.g., "imho" , "☺"), length-based features (e.g., average sentence length), position information, and function words. When present, certain Web structures (e.g., Web links, citations within documents, and "respond-to" relationships) may also be good opinion evidence. For example, research on opinion in the political domain has shown that links tend to be less noisy than

text-based opinion evidence (Agrawal, Rajagopalan, Srikant, & Xu, 2003; Efron, 2004; Malouf & Mullen, 2008). However, style-based opinion-bearing features are sensitive to the target data domain and are usually not strong opinion indicators in the blogosphere, where Internet users often write in "free-style".

Since each source of opinion evidence has its own characteristics and captures different aspects of opinions, opinion-bearing features from more than one source of opinion evidence are often used for opinion detection. Most studies have suggested that a fusion of various opinion-bearing features surpasses the use of any single subset of features (Chesley et al., 2006; Gamon, 2004; Hatzivassiloglou & Wiebe, 2000; Y. Niu, et al., 2005; Riloff, Wiebe, & Wilson, 2003; Wiebe, 2000; Wiebe et al., 1999; Wiebe, Wilson, & Bell, 2001; Yang et al., 2007).

2.2.1 Ad Hoc Rule-Based Opinion Detection

The basis of the *ad hoc rule-based approach* is the opinion lexicon, which is a list of the terms or patterns that provide evidence for the presence of opinion or fact. High-quality knowledge-based opinion-bearing features normally provide the basis for an opinion lexicon, and statistical and structure-based opinion-bearing features are usually added to the opinion lexicon after manual inspection.

The most naïve yet effective way of using an opinion lexicon is to apply a simple matching rule that assumes the presence of one or more opinion-bearing

features that can serve as a proxy for the opinion class label[5]. For example, in sentence-level subjective classification (Hatzivassiloglou & Wiebe, 2000; Wiebe, 2000; Wiebe & Wilson, 2002), a sentence would be classified as subjective if at least one member of a set of adjectives occurred in the sentence; if not, it would be classified as objective. Instead of providing a binary judgment based on simple match, an opinion score can be calculated for the target unit by counting total number of occurrences or summing the scores of opinion-bearing features (Gamon & Aue, 2005; Kim & Hovy, 2005), which are sometimes normalized by unit length (Eckle-Kohler, Kohler, & Mehnert, 2005). In order to handle negations and other exceptions, it is common to introduce other rules in addition to simple matching or scoring rules. For example, one rule can define that, if 'not' occurs in a piece of text, the semantic orientation of this text unit is to be reversed.

Although manually created opinion lexicons are of high-quality, they have low coverage. In order to automatically expand the initial opinion lexicon (i.e., the seed set), two approaches have been proposed. One approach requires knowledge-intensive resources such as an online dictionary (e.g., WordNet) or thesaurus (i.e., the dictionary-based approach), while the other learns opinion-bearing terms from a large collection of labeled or partially labeled data (i.e., the corpus-based approach). As demonstrated in Figure 1, both approaches start with a few seed words: (1) the similarity between a new word and each of the seeds is calculated, based either on semantic word relations or on co-occurrence statistics[6]; (2) the

[5] Each feature is sometimes viewed as one sub-rule.
[6] Specific similarity measures can be found in Hatzivassiloglou and McKeown (1997), Kanayama and Nasukawa (2006), Turney and Littman (2003, 2004) and Wiebe et al. (2004).

new word is labeled with the opinion class of the most similar seed word and added to the seed set; and (3) the whole process is repeated until there are no new words. This bootstrapping process, which has proven successful in expanding opinion lexicons (Riloff & Jones, 1999; Riloff et al., 2003; Thelen & Riloff, 2002), exemplifies a term-level opinion detection strategy that requires only a few labeled data.

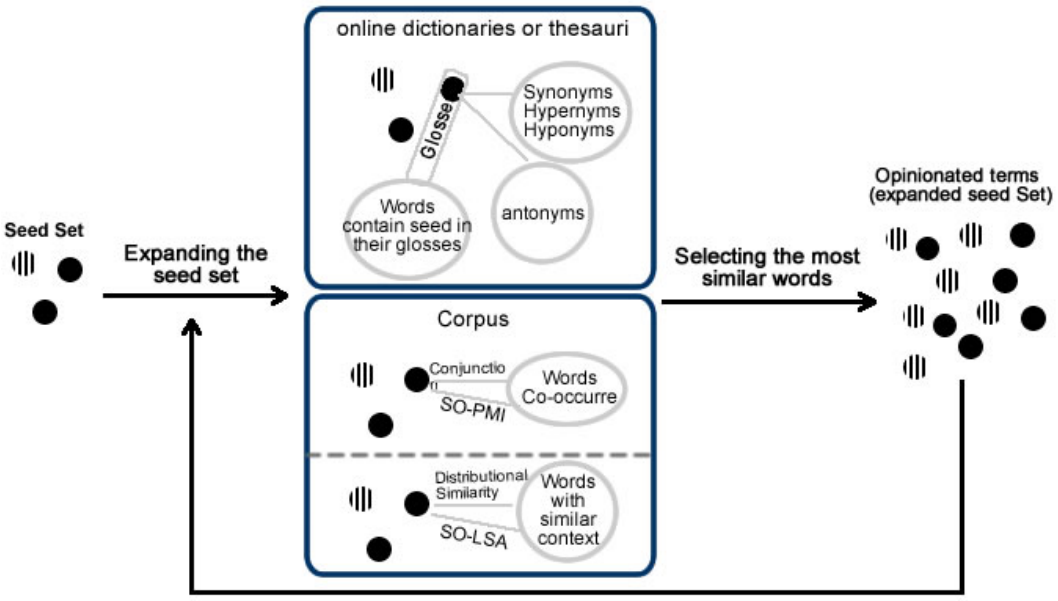

Figure 1. Bootstrapping procedure for expanding the opinion lexicon in both dictionary-based and corpus-based approaches. Dots with different shapes belong to different opinion class.

The strength of the ad hoc rule-based approach lies in the fact that it is conceptually intuitive and easy to implement. However, this strength comes with four limitations: (1) a heavy dependence not only on the quality of opinion-bearing features but also on the quantity and diversity of features necessary to

capture the variety of features used in opinion expressions; (2) the need for sophisticated strategies such as linguistic-parsing to extract opinion-bearing features and for expert knowledge and man-power to implement, maintain, and update complex rule-based systems; (3) an inability to capture all exceptions; and (4) the absence of obvious opinion-bearing features in some opinion expressions. As a result, the ad hoc rule-based approach is difficult to scale up to deal with the real world collection of texts on the Web.

2.2.2 Machine Learning-Based Opinion Detection

The machine learning approach is more practical in opinion detection than the ad hoc rule-based approach due to its fully automatic implementation and its ability to handle large collections of Web data. Supervised learning is a mature and successful solution in traditional topical classification and has been adopted and investigated for opinion detection with satisfactory results (Wiebe et al., 2004; Yu & Hatzivassiloglou, 2003; Zhang & Yu, 2007). Semi-supervised learning is a promising direction for topical classification, but it has been investigated in only a few opinion detection studies.

2.2.2.1 Some Important Supervised Classification Algorithms

A classification algorithm is basically a function (f: X->Y, or P(Y|X)) for assigning the class label (Y) to a test example (X). A supervised classification algorithm learns the parameters of this function from the labeled data. A comprehensive survey by Sebastiani (2006) summarizes important supervised

classification algorithms: Naïve Bayes, a generative classifier that estimates prior probabilities of P(X|Y) and P(Y) from the training data and "generates" the posterior probability of P(Y|X) based on these prior probabilities; Support Vector Machine (SVM), a discriminative classifier that makes no prior assumptions based on the training data and directly estimates P(Y|X); and the lazy learning algorithm K-Nearest Neighbors (KNN), which does not require prior construction of a classification model. In both topical and opinion classification, Naïve Bayes and SVM are the most common and effective supervised learning algorithms.

Naïve Bayes Classifier

A Naïve Bayes classifier predicts a class label based on prior probability calculated under the simplifying assumption of term independence: The use of a term in a document does not depend on any other term or terms in the document. Although the assumption of term independence does not hold in the real world, a simple Naïve Bayes classifier is surprisingly efficient at topical text classification.

According to the basic Bayesian formula, the probability that document d (t_1, t_2, … , t_n) belongs to category C_j is determined as follows:

$$
\begin{aligned}
P(C_j|\, d) &= P(C_j)P(d|C_j) \,/\, P(d) \\
&= P(C_j)P(d|C_j) \\
&= P(C_j)P(t_1, \dots, t_n|C_j) \\
&= P(C_j)P(t_1|C_j)P(t_2, \dots, t_n|C_j, t_1) \\
&= P(C_j)P(t_1|C_j)P(t_2|C_j, t_1) \dots P(t_n|C_j, t_1, t_2, t_3, \dots t_{n-1}) \qquad (1)
\end{aligned}
$$

where P(d) is ignored because it does not affect the probability computation across categories (j=1…m). Based on the assumption of term independence, formula (1) can be simplified as:

$$P(C_j|\, d) = P(C_j)P(t_1|C_j)P(t_2|C_j) \ldots P(t_n|C_j) \qquad (2)$$

where

$P(C_j)$ = # of documents belonging to C_j / total # of documents

and

$P(t_k|C_j)$ = probability of a term k occurring in category j

= # of documents in C_j with term k / total # of documents in C_j

Support Vector Machines

SVMs are currently the most effective text classification algorithms (Sebastiani, 2006); they are the favored supervised learning method for opinion classification because of their consistently robust performance in natural language processing (Joachims, 2001). The key to a binary SVM is to find a decision surface in the feature space that will separate positive and negative training examples. In the case of opinion detection, this means separating opinions from non-opinions. Based on the intuition that the classification decision is less uncertain and has good generalization capability when there are no examples near the decision surface, the best decision surface is one with maximal margin to training examples (see the left-hand example in Figure 2). When examples are not linearly separable, a kernel function Φ is sometimes used to map the original feature space onto a new space (see the middle and right-hand examples in Figure

2). In order to classify large-scale Web data more effectively, SVMs with an optimized learning algorithm (e.g., Sequential Minimal Optimization, or SMO) have sometimes been employed (Abbasi et al., 2008; Gamon, 2004; Y. Niu et al., 2005; Whitelaw Garg & Argamon, 2005a). A detailed treatment of the application of these models to text classification can be found in Joachims (2001), and a simple procedure for using an SVM for reasonable results can be found in Hsu, Chang, and Lin (2003).

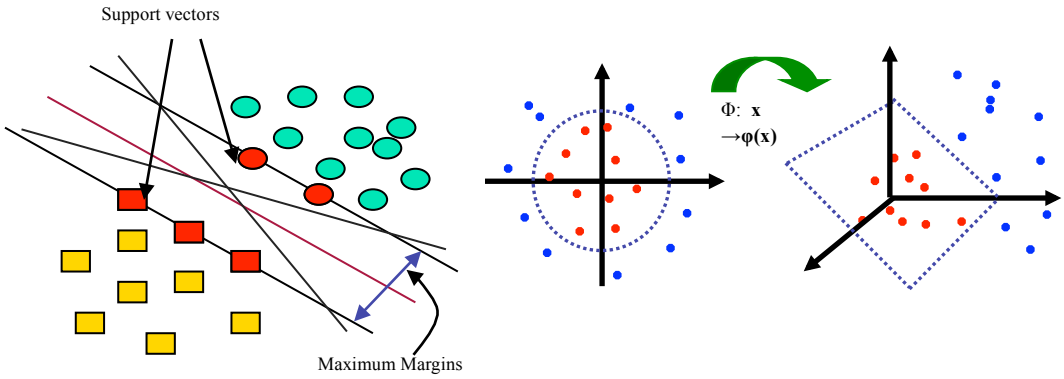

Figure 2. SVM maximum margins and kernel functions.

2.2.2.2 Supervised Opinion Detection

With no classification techniques developed specifically for opinion detection, state-of-the-art topical supervised classification algorithms are often tailored for the task of opinion detection. Normally, this is done in the following manner:

1. Instead of the ordinal frequency of features, binary values (presence/absence) are used to represent the classifying unit. This is motivated by the extreme brevity of the text unit when classifying short

Web documents such as movie reviews (Pang & Lee, 2008; Pang et al., 2002) or, more generally, when classifying opinions at the sentence level. The preference for binary values may be due to the characteristics of opinion detection, where occurrence frequency is less influential (i.e., a single occurrence of opinion evidence is sufficient);

2. Because an opinion is subtler than a topic, a wider variety of evidence (e.g., linguistic features, links) is investigated in addition to auto-generated features (e.g., bag-of-words, n-grams) (Gamon, 2004; Mullen & Collier, 2004; Y. Niu et al., 2005; Pang & Lee, 2004; Whitelaw et al., 2005a; Yu & Hatzivassiloglou, 2003);
3. Exhaustive parameter optimization is not usually performed for classification models. Instead, default parameter values from an off-the-shelf toolkit are generally used with or without preliminary analysis of the effect of varying parameter values (Cui et al., 2006; Mullen & Collier, 2004; Ng et al., 2006; Pang et al., 2002; Whitelaw et al., 2005ab; Zhang et al., 2007). The under-utilization of parameter optimization indicates the strong impact of opinion-bearing features.

Supervised language models that take into account the sequence or structure of text have also been applied in opinion detection (Conrad & Schilder, 2007; Jurafsky & Martin, 2008; McDonald, Hannan, Neylon, Wells, & Reynar, 2007; Pang & Lee, 2004). Such statistical models of word sequences are generalized in the context of opinion detection to predict the opinion class of a given text unit based on previously labeled text units. The concepts in language models are easy

to follow, but the construction of such a model is complex and its execution is slow when the algorithm needs to backtrack to check the opinion class assigned either to the entire document or to one or more previous sentences.

Machine learning produces promising results when multiple features are used in conjunction. In most cases, SVMs have shown marginal improvement over Naïve Bayes classifiers. Cui et al. (2006) have argued that discriminative classifiers such as SVMs are more appropriate for sentiment classification than generative models because they can better differentiate mixed sentiments (i.e., both positive and negative words are used in the same review). However, when the set of training data is small, a Naïve Bayes classifier might be more appropriate since SVMs must be exposed to a large set of data in order to build a high-quality classifier.

Models based on supervised learning have shown advantages in opinion detection on the Web due to the fully automatic construction and implementation of classifiers; the ability to handle large and noisy Web data; performance comparable to complex linguistic approaches when incorporating different types of features; and the ease of access to tools with built-in classification algorithms. The biggest limitation associated with supervised learning is that it is sensitive to the quantity and quality of the training data and may fail when training data are biased or insufficient. Opinion detection at the sub-document level raises additional challenges for supervised learning based approaches because there is little information for the classifier.

2.2.2.3 Semi-Supervised Opinion Detection (Bootstrapping[7])

In contrast with supervised learning, which learns from labeled data only, semi-supervised learning (SSL) learns from both labeled and unlabeled data. SSL is a relatively new machine learning approach to opinion detection, motivated by the lack of labeled data in real world applications. The main idea behind SSL is that, although unlabeled data hold no information about classes (e.g., "opinion" and "non-opinion"), they do contain information about joint distribution over classification features. Therefore, when there is limited labeled data in the target data domain, using SSL with unlabeled data can achieve improvement over supervised learning.

The bootstrapping procedure for extracting opinion-bearing features depicted in Figure 1 is a simple form of SSL algorithm, known as self-training[8], in which "a tagger is retrained on its own labeled cache on each round" (Clark, Curran, & Osborne, 2003). One shortcoming of self-training is that the resulting data may be biased by the opinion detector: That is, the final set of labeled data may be made up of those examples which are easiest for this particular opinion detector to identify. Next section reviews more sophisticated SSL algorithms which may make better use of limited labeled data and further reduce classification errors.

[7] Although bootstrapping is often used in the literature of opinion mining to refer to the problem setting of SSL, SSL will be used in this paper because it reflects the nature of the learning algorithm.

[8] Self-training, also known as mutual bootstrapping or self-teaching, is conceptually equal to the pseudo relevant feedback technique in information retrieval where the top n retrieved results to a given query are assumed to be relevant and are used to form a new query.

2.3 Common Semi-Supervised Learning (SSL) Algorithms

SSL overcomes the insufficiency of labeled data by making use of unlabeled data. Beginning with a limited set of labeled data, semi-supervised learning (SSL) applies an iterative process of automatically labeling unlabeled data and then selecting auto-labeled data to replenish the labeled dataset. According to a survey of SSL by Zhu (2008), the most commonly used SSL algorithms include self-training, Expectation-Maximization (EM) with generative mixture models, co-training, Semi-Supervised Support Vector Machines (S^3VMs), and graph-based methods.

2.3.1 Self-Training

Self-training is the simplest SSL method because it requires only one classifier to automatically label unlabeled data. The major steps in self-training are: (1) train an initial classifier on the labeled dataset; (2) apply this classifier to the unlabeled data and select the most confidently labeled data as determined by the classifier to augment the original labeled dataset; and (3) re-train the classifier by repeating the whole process from step (1). Simple pseudo code for self-training is illustrated in Figure 3.

Self-training is easy to apply. It is a wrapper approach that can be applied to any existing system as long as a confidence score can be produced. Self-training keeps a system in a black box and avoids dealing with any inner complexities. The simplicity and flexibility of self-training is attractive for modifying current

opinion detection systems, especially those based on complicated linguistic analysis. The downside of self-training, however, is that there is no guarantee of its performance. If mislabeled examples are selected and added to the labeled dataset, the classifier trained with a self-labeled dataset will be likely to choose more examples like these mislabeled ones. Therefore, if errors occur during self-training, they will be reinforced, especially if the errors occur in the early stages of the iterative process.

Input: classifier *C*, a small set of labeled data *L*, a large amount of unlabeled data *U*

Loop until iteration procedure converges or loop for *k* iterations

1) Train *C* on *L;*
2) Apply trained *C* to *U*;
3) Select top *n* results returned by *C* and add them back to *L;*

Output: Extended *L* and an updated *C*

*Figure 3***.** Bootstrapping procedure in self-training.

Riloff and Wiebe (2003) and Riloff et al. (2003) have successfully applied self-training to sentence-level opinion detection. The initial classifier *C* is either a simple rule-based classifier built using a few manually created opinion seeds or a supervised classifier trained on a few manually labeled data. After bootstrapping a simple high-precision classifier through several iterations, the process generates a large labeled dataset and/or rich opinion lexicon, either of which can then be used by any supervised opinion detection algorithm or lexicon-based opinion detector. Across several experiments carried out by Wiebe and Riloff (2005), a self-trained

Naïve Bayes classifier using this procedure achieved the best recall with modest precision when classifying subjective sentences.

In the literature of opinion detection, self-training has made positive contributions in dealing with three types of data domain: movie reviews, news articles and blog posts. However, more research is needed to confirm the effectiveness of self-training for opinion detection.

2.3.2 Expectation-Maximization-Based SSL

The Expectation-Maximization (EM) algorithm was introduced by Dempster, Laird and Rubin (1977). EM refers to a class of iterative algorithms for maximum-likelihood estimation when dealing with incomplete data. Nigam, McCallum, Thrun and Mitchell (1999) combined the EM algorithm with Naïve Bayes classifier to resolve the problem of topical classification by treating unlabeled data as incomplete data. Strong assumptions underlie the integration of EM learning with text classification: All data are generated by a mixture model which holds a one-to-one correspondence between mixture components and classes (i.e., two models exist to generate the opinion and non-opinion data, respectively), and this mixture model can be learned if there is a large amount of data, labeled or unlabeled, that approximates the real data distribution.

EM-based SSL is as a special form of self-training under the mixture model assumption. Each iteration in EM-based SSL involves the following steps: (1) train the Naïve Bayes classifier with labeled data; (2) apply the Naïve Bayes classifier to each unlabeled document to assign a "probabilistically-weighted"

class label (Nigam et al, 1999, p. 104) (i.e., the Expectation or E-step); (3) retrain the Naïve Bayes classifier with both the originally labeled and the unlabeled data to maximize the posteriori estimate for the classification parameters (i.e., the Maximization or M-step); and (4) repeat steps 2 through 3 until the Naïve Bayes classifier does not change. In the end, EM finds the local maximization of the likelihood of the classification parameters given all the data.

When the amount of unlabeled data far outnumbers the amount of labeled data, the EM learning process can be understood as unsupervised clustering with a few labeled data to provide information about the class label. Under the mixture model assumptions, the dataset should contain two tight clusters: one corresponding to the opinion examples and the other corresponding to non-opinion examples. But the real world is more complicated, and this assumption is usually violated. Nigam et al. (1999) proposed two extensions to EM to deal with violated assumptions. The first extension introduced a new weighting parameter λ for controlling the influence of unlabeled data (EM-λ). Since the data collection may contain clusters that do not correspond to opinion classes, a small λ can emphasize the clusters that are favorable to the labeled data. The second extension introduced a many-to-one correspondence between mixture components and classes to relax the one-to-one assumption. The extended EM algorithm was shown to achieve significant improvement in three real-world topical classification tasks. Detailed algorithms and a theoretical analysis can be found in Nigam et al. (1999).

To investigate how EM works, Nigam et al. (1999) identified the top features used by a classifier on each iteration when classifying a course homepage. They observed that more general features such as "handout", "problem" and "homework" appeared among the top features during the self-training process, indicating that EM is able to reduce the specialty of using a few labeled data by bringing out more generalized features in a large amount of unlabeled data.

Motivated by the success of combining EM and Naïve Bayes in text classification tasks, a few studies have used the same SSL strategy to deal with the absence of large amounts of labeled data for polarity detection[9]. The EM-NB SSL algorithm has yielded better performance than either supervised approaches or unsupervised lexicon-based approaches in sentiment classification at various levels and with different data domains, including blog data (Aue & Gamon, 2005; Gamon & Aue, 2005; Gamon et al., 2005; Takamura, Inui, & Okumura, 2006). Takamura et al. (2006) found that the EM-NB SSL algorithm always improved the classification accuracy regardless of the quantity of labeled data (varying from 100 to 1000); it produced better results with more unlabeled data; and it extracted more contextual features when compared with the top 100 features used in the initial and final Naïve Bayes model.

The consistently positive performance of EM-based SSL in topical classification and polarity detection suggests that it may be useful in opinion detection. However, EM-based SSL is limited by the mixture model assumption:

[9] Theoretically, any generative classifier can be combined with EM learning. Naïve Bayes is often selected because it offers a probabilistic foundation for EM and is efficient with large-scale datasets.

If the natural clusters in the target data domain are different from the target classes, EM-based SSL can hurt performance since EM learning was not designed with classification tasks in mind. In the case of opinion detection, both opinion and non-opinion data are often a mix of opinion and non-opinion vocabularies, and it is not clear whether there is an underlying cluster structure.

2.3.3 Co-Training

Co-training, also known as collaborative bootstrapping, assumes that redundancy exists in the target data domain, and thus more than one view of the data can be used to classify each example. Two assumptions supported the original co-training algorithm defined by Blum and Mitchell (1998): (1) there exist two views for each target example, and these two views are independent (independence assumption); and (2) data distribution is compatible with the target function, and each view classifier (i.e., the classifier trained on one view of the data) alone can produce the right prediction label for most examples (compatibility assumption). For example, when classifying Web pages, words appearing either on the Web page or on its incoming links are sufficient to build a view classifier. Pseudo code for the original co-training algorithm is provided in Figure 4.

When labeling new documents, a combined classifier C will be constructed by multiplying the predictions of the two final classifiers C_1 and C_2[10]. In the "course home page" classification experiments conducted by Blum and Mitchell

[10] After multiplying the predictions, the class probability scores are sometimes normalized so that they sum to one (Nigam & Ghani, 2000).

(1998), this co-training algorithm successfully eliminated more than half of the classification error generated by the supervised learning algorithm. In this preliminary experiment, only 12 labeled Web pages (three positive, nine negative) were selected and 776 Web pages were treated as unlabeled data, demonstrating that co-training could be helpful in reducing the need for labeled data.

One positive characteristic of co-training is that disagreement between the two view classifiers sets the upper bounds for the error rate. Because the two view classifiers are expected to agree with each other under the independence and compatibility assumptions, a low error rate can be obtained by co-training. The mathematical proof and theoretical framework for co-training can be found in Blum and Mitchell (1998) and Abney (2002).

There are revised versions of co-training that are based on more practical and less restrictive co-training assumptions that correspond to different co-training parameters: building the C_1 and C_2 classifiers; selecting auto-labeled data; reducing disagreement between C_1 and C_2; and making decisions regarding u and k.

Input: Classifier C_1, C_2, a small set of labeled examples L,
and a large number of unlabeled examples U

Initiate parameters: $u = 75$, $p = 1$, $n = 3$, $k = 30$

Randomly select u examples from U to create a pool U'

Loop for k iterations

1) Train a view classifier C_1 with only the x_1 aspect of each example x in L
2) Train a view classifier C_2 with only the x_2 aspect of each example x in L
3) Apply C_1 to U' to label p positive and n negative examples with most confidence
4) Apply C_2 to U' to label p positive and n negative examples with most confidence
5) Add the newly labeled examples to L
6) Randomly select $2p+2n$ examples from U to add to L

Output: Expanded L and two trained classifiers C_1 and C_2

Figure 4. Co-training algorithm defined in Blum and Michell (1998).

2.3.3.1 Building C_1 and C_2

The essential factor for applying the original formulation of co-training is to find a natural split in the feature set and then build two classifiers, C_1 and C_2, each based on a different view (e.g., in image classification, each image can be naturally represented by both its text description and its visual attributes (Feng & Chua, 2003)). For some NLP applications, content and contextual features are often treated as separate views (Collins & Singer, 1999; Pierce & Cardie, 2001). In the case of opinion detection, however, it is hard to find this kind of natural

split without domain knowledge. Fortunately, relaxing the co-training assumptions of independence and compatibility and exploiting redundancy in the feature set seems to suffice for co-training (Zhang, 2004). In their comprehensive evaluation of co-training, Nigam and Ghani (2000) state that, when there is no natural split, randomly splitting the feature set will still outperform regular algorithms using a single feature set (e.g., EM) as long as there is sufficient redundancy in the feature set. Since the opinion-bearing feature set is usually large and diverse, it is reasonable to assume the existence of a certain level of redundancy.

When there is no natural split in the feature set, C_1 and C_2 classifiers can be built without splitting the feature set. To prepare C_1 and C_2 for co-training, Goldman and Zhou (2000) used two different supervised learning algorithms; Y. Zhou and Li (2005) applied two different parameter configurations of the same learning algorithm; Maeireizo, Litman and Hwa (2004) trained two classifiers for each of the target classes; and Jin, Ho and Srihari (2009) created disjoint training sets t1 and t2 on each iteration. By selecting the labeled sentences agreed upon by both C_1 and C_2, Jin et al. (2009) successfully applied the co-training algorithm to identify opinion sentences in camera reviews. For different camera products, this co-training approach always increased classification precision with little or no negative effects on recall.

Wang and Zhou (2007) derived a theorem to explain the success of co-training studies without feature splitting: The key to the success of co-training is

the existence of two largely different initial learners, regardless of whether they are built by using two views, two learning algorithms, or any other channel. The two different learning algorithms can be understood as providing different views of the same example based on their own assumptions and biases. Thus the phrase "view classifier" will be used to describe C_1 and C_2 in the rest of the paper.

2.3.3.2 Selecting auto-labeled data

The original co-training algorithms select data with labels assigned by the C_1 and C_2 view classifiers and add them to the labeled dataset L. The newly labeled data from C_1 and C_2 is referred to as "auto-labeled data".

Under relaxed co-training assumptions, more sophisticated strategies are applied for selecting auto-labeled data. Ideally, the most informative and useful auto-labeled data should be selected. The usefulness of an example is related to information entropy: The more uncertain a classifier is about an example, the more value this example has for the classifier. Pierce and Cardie (2001) passed the most uncertain auto-labeled data to human annotators before adding them back to the labeled dataset and retraining the classifiers.

Cao, Li and Lian (2003) proposed a co-training algorithm where, for each iteration, each view classifier would ask the other classifier to label those examples about which the first classifier was most uncertain. The results of this justified co-training algorithm was compared with the results of several previous co-training studies as well as with one self-training study. All of the results were

promising when a natural split did not exist in the feature set, the new co-training algorithm showed significant improvement over older algorithms: When there was a natural split in the feature set, the new co-training algorithm performed almost as well as older algorithms; and, when the feature set was randomly split, the new co-training algorithm significantly outperformed both the self-training algorithm and the original co-training algorithms.

Feng and Chua (2003) adopted a fusion approach that involved both automatic and manual selection of auto-labeled data and improved the performance of image annotation by 10% on the F measure[11] as compared to the supervised learning approach. There is also a revised co-training algorithm known as tri-training that replaced the role of human annotators with a third classifier (Z. H. Zhou & Li, 2005). Tri-training was a "majority teach minority" approach: If two learners agree on the label for one example, this example will be used to teach the third classifier. This group of revised co-training algorithms places fewer or no constraints on the view classifiers and is often treated as a separate algorithm from co-training. Due to the complexity involved in using more than two classifiers, this strategy was not examined in the current study.

2.3.3.3 Reducing disagreement between C_1 and C_2

Despite the compatibility assumption, the original co-training algorithm makes no explicit effort to reduce disagreement between the view classifiers C_1 and C_2. However, several subsequent studies on co-training (e.g., Collins &

[11] F measure combines precision (P) and recall (R) and is calculated as F = 2 * ((P * R) / (P + R)).

Singer, 1999) have suggested that it is beneficial to do so. The motivation behind reducing disagreement between C1 and C2 is to lower the error rate of co-training since the disagreement rate sets the upper bound for the error rate. Collins and Singer (1999) successfully used co-training to resolve an NLP problem by minimizing mismatch between C_1 and C_2. This was achieved by forcing them to agree with each other on predictions for labeled data and to agree with each other as much as possible on predictions for unlabeled data. Abney (2002) applied a greedy agreement algorithm for each iteration in the co-training process, and this approach performed equally well.

Instead of providing the entire unlabeled dataset U for labeling by view classifiers C_1 and C_2, use of a smaller pool u was suggested by Blum and Michell (1998). The rational was that a small u could indirectly force C_1 and C_2 to select uncertain, albeit useful, examples and to avoid selection of their preferred examples only. Wang and Zhou (2007) pointed out, that the larger the value of u, the smaller the difference between C_1 and C_2 and the less that can be learned during co-training.

Co-training continues until C_1 and C_2 converge or for k iterations, where k is usually decided arbitrarily. Several studies have concluded that, after several iterations, the performance of co-training will make no further improvement and may actually begin to degrade (Pierce & Cardie, 2001; Wang & Zhou, 2007). The explanation for this phenomenon is that C_1 and C_2 become more and more similar as co-training proceeds; and, at a certain point, there is nothing valuable to be

learned from each other. Furthermore, since C_1 and C_2 are combined for testing, their classification errors will be enhanced when they are very alike. Wang and Zhou (2007) therefore proposed an estimation value for k so that the co-training process would stop at or near the point of best performance based on measures of similarity between C_1 and C_2.

2.3.3.4 Summary

Although co-training can be understood as a special type of self-training that loops over a compound classifier with a complex inner structure, it has advantages over self-training: Co-training requires less labeled data than self-training since each labeled example is used twice; co-training converges faster than self-training; and, when there are different views for the target examples, co-training is conceptually clearer than self-training, which simply mixes features. However, co-training has only been mentioned as a future direction or as an unfeasible SSL algorithm for opinion detection because of the lack of natural feature splits (Suzuki, Takamura, & Okumura, 2006; Wiebe et al., 2001; Wiebe & Riloff, 2005).

2.3.4 Semi-Supervised Support Vector Machines (S^{3}VMs)

Originally known as Transductive Support Vector Machines (TSVMs), semi-supervised support vector machine S^{3}VM (Bennett & Demiriz, 1998) is a better name for this group of learning algorithms since they are not only capable of transduction but of induction as well (Chapelle, Schölkopf & Zien, 2006; Zhu,

2008). S^3VMs are a natural extension of support vector machines (SVMs) in the semi-supervised spectrum and are designed to find the maximal margin decision boundary in a vector space containing both labeled and unlabeled examples. In Figure 5, black dots correspond to unlabeled examples and black circles with positive or negative signs correspond to labeled examples. While traditional SVMs draw a boundary, indicated by the solid lines, to separate labeled examples, S^3VMs draw boundaries, indicated by the dashed lines, to separate examples so as to keep positive and negative examples apart. Mathematical descriptions for S^3VMs can be found in Bennett and Demiriz (1998). Although there are several suggestions for S^3VM optimization, S^3VM implemented in SVM^{light} (Joachims, 1999a) and based on local search is commonly adopted.

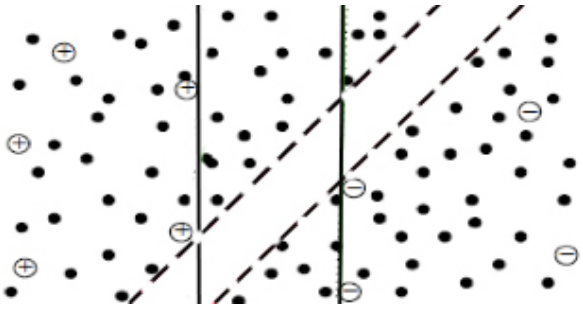

Figure 5. A visual representation of S^3VMs modified from Zhu (2008).

When the labeled training dataset is small and the unlabeled test dataset is large, S^3VMs have outperformed SVMs in a variety of topical classification tasks (Chapelle, Weston, & Scholkopf, 2003; Joachims, 1999b; Tong & Koller, 2001) as well as in some NLP applications (Goutte, Déjean, Gaussier, Cancedda, & Renders, 2002). Although SVMs are the most favored supervised learning method

for opinion detection (Cui et al., 2006; Pang et al., 2002; Zhang & Yu, 2007), S^3VMs have not yet been applied for opinion detection or its related tasks[12].

The advantages of utilizing SVMs for opinion detection lie in its ability to handle a mix of different types of features and to work with diverse Web content (Gamon, 2004; Y. Niu et al., 2005). The drawbacks of SVM are its computing inefficiency when contrasted with Naïve Bayes classifiers and its degraded performance when there are not enough labeled data (Lin, Wilson, Wiebe, & Hauptmann, 2006). S^3VMs inherit the advantages of SVM while overcoming the negative impact of limited labeled data. Therefore, when an existing opinion detection system already uses SVM, it is reasonable to select S^3VMs as the SSL algorithm.

2.3.5 Graph-Based SSL

A large group of SSL algorithms falls into the category of graph-based algorithms, which rely on the geometry of both labeled and unlabeled data. Building a graph is straightforward: Each example is represented as a node in the graph, and related examples are linked though weighted edges, usually based on similarity measurements. The absence of an edge between two nodes stands for infinite distance. Ideally, the class labels of labeled examples will propagate through the edges to label all unlabeled examples. How to construct the graph is

[12] Dasgupta and Ng (2009) used S^3VMs as part of their evaluation of unsupervised and interactive polarity classification for reviews. Their system gained the same benefit from collecting user feedback and from applying S^3VMs with limited labeled data.

as important as, if not more important than, the estimation of a prediction function on the graph (Zhu, 2008).

Graph-based SSL has often been adopted for the task of polarity detection. One group of studies mapped links among blog posts to the edges on the graph (Agrawal et al., 2003; Kale et al., 2007; Malouf & Mullen, 2008). Another group of studies used graph-based learning to label polarity terms by building graphs that linked unlabeled and labeled terms via certain lexical relations (Hatzivassiloglou & McKeown, 1997; Kamps, Marx, Mokken, & Rijke, 2004). Graph-based learning has also been applied for opinion detection. An influential graph-based application for subjective analysis was developed by Pang and Lee (2004), who extracted subjective sentences from text using a graph minimum cuts algorithm. Their graph captured the knowledge of individual sentences (i.e., the probability score that a sentence was subjective) as well as pair-wise associations between sentences (i.e., estimations of how important it was to link two sentences based on their distance in the text).

Using a graph to describe a problem space is straightforward and intuitive; and the relations between labeled and unlabeled data are more clearly presented by a graph than by any other SSL algorithms. However, graph-based SSL is not ideal for dealing with large-scale data because it requires pair-wise calculations between all data, and the whole graph needs to be recalculated every time a new

example must be classified[13]. Furthermore, an increase in the distinct regions on a graph requires more labeled data to predict class labels for each region (Bengio, Delalleau & Roux, 2006).

2.4 Major Challenges and Current Solutions in Opinion Detection

Challenges for opinion detection are rooted in the subtle nature of opinions, which makes them more complex than facts and heavily dependent on the context in which they are used. Noisy Web data can also generate additional challenges for opinion detection.

2.4.1 Context Sensitivity

The subtlety of opinion expression is directly reflected in its high sensitivity to context. The same expression may be an opinion in one context and a non-opinion in another. For example, "Bush is an actor" could be a purely objective statement in a review of a movie starring an actor named Bush. However, it could be a negative opinion if used in the context of a presidential election. Context dependency is more noticeable in the case of polarity detection. The word "great" is normally associated with positive opinions, but the sentence "It is just great" may actually express a negative opinion. Therefore, when judging an opinion expression, where and how an expression is delivered may be as important as, if not more important than, what is delivered.

[13] Although it is common practice to fix the graph on the initial labeled and unlabeled data to avoid the expenses of re-computing the whole graph every time a new data point is encountered, it becomes less accurate when too many new data have been added to the graph.

Currently, there is no effective solution for processing contextual information, and sophisticated linguistic analyses are usually required. However, large amounts of opinion-labeled data can better address the challenges of context sensitivity by producing superior high order n-grams (Cui et al., 2006), which is the simplest way to retain context for words.

2.4.2 Domain Dependency

Domain dependency may seem less problematic for opinion detection than topical classification since generic opinion-bearing words such as "good" and "bad" are not limited to any particular domain. But there are few generic opinion-bearing words, and it is therefore necessary to extract opinion-bearing features from the target data collection. These features are generally domain dependent and may not be reusable in another domain for several reasons: (1) there are specific opinion-bearing words associated with different domains (e.g., "cheap" and "long-lasting" are frequently used in product reviews, but not in movie reviews); (2) different domains have different stylistic expectations for language use (e.g., news articles are less likely than blogs to use words such as "crappy" or "sooooooooo"); and (3) some opinion-bearing words can be either positive or negative depending on the object (e.g., "small" may be positive in "a small camera" but negative in "a small memory card"). Since information used for opinion detection is typically lexical and lexical means of expressing opinions may vary not only from domain to domain but also from register to register, an

opinion detection strategy that works for one target data domain generally will not work for another data domain.

Most opinion detection systems borrow opinion-labeled data directly from non-target data domains when there are few opinion-labeled data in the target domain or when the characteristics of the target domain make it difficult to detect opinions if the non-target data appear to be "relevant to the application and of sufficient quantity" (Conrad & Schider, 2007, p. 235). This approach is especially common in opinion detection in the blogosphere. For example, Chesley et al. (2006) leveraged blog training data with non-blog training data containing relatively "pure" opinion information; and most participants in TREC's Blog track have crawled the Web to generate a large amount of opinion-labeled training data. However, according to Aue and Gamon (2005), who compared four strategies for utilizing opinion-labeled data from one or more non-target domains, using non-target labeled data without an adaptation strategy is not as efficient as using labeled data from the target domain, even when the majority of labels are assigned automatically by a self-training algorithm.

Similar domain adaptation strategies have been investigated for sentiment analysis. Blitzer, Dredze and Pereira (2007) proposed a structural correspondence learning (SCL) algorithm for sentiment classification to reduce the classification error of a classifier trained with non-target data. The key to this domain adaptation strategy is to implicitly associate domain specific features in the target and non-target data domains with certain general features that are used frequently

in both domains and are relevant to the opinion class. As a result, even if a feature in the target domain has never occurred in the non-target domain, the class label can be predicted by looking up its corresponding feature(s) in the non-target domain.

To generate general features, both labeled and unlabeled data in the non-target domain and unlabeled data in the target domain are needed. This approach is sometimes referred to as a semi-supervised domain adaptation strategy.

A study by Tan, Cheng, Wang and Xu (2009) made use of general features in both the target and non-target domains to address the domain adaptation problem in sentiment analysis. Their approach differed from the study by Blitzer et al. (2007) in that only labeled data in the non-target domain were used with an SSL algorithm that put more weight on target data for sentiment classification. Regardless of their positive contributions to sentiment analysis, both of these domain adaptation strategies involve sophisticated and expensive methods for selecting general features and applying them to sentiment analysis.

2.4.3 Informal and Noisy Web Content

Malouf and Mullen (2008) pointed out that "the development of interactive 'Web 2.0' is changing the nature of typical Web texts and has raised significant new challenges for natural language" (p. 1). Unlike professionally edited newspapers or traditional Web sites that follow certain conventions of editing, blog posts are generally informal and typically conversational and non-standard

(Malouf & Mullen, 2008). The use of informal language poses new problems for linguistic analysis and information retrieval techniques when applied to opinion detection. In an attempt to capture some of the informal language used on the Web, an opinion acronym lexicon consisting of a manually filtered subset of chat acronyms and shorthand text messages from NetLingo[14] (e.g., "imho"="in my humble opinion") was developed by Yang et al. (2007).

Like many documents published on the Web, blog data is noisy in that it contains non-content peripheral materials such as navigation links and advertisements (Glance et al., 2005), spam blogs (splogs) or post spam comments, and duplicated content. In 2007, Technorati tracked between 3,000 and 7,000 new splogs created everyday (Sifry, 2007). When there are opinion-bearing features or opinion expressions in peripheral materials and spam, they can mislead the opinion detection system and reduce its performance. Several studies in TREC's Blog track (Macdonald, Ounis, & Soboroff, 2007; Ounis, Rijke, Macdonald, Mishne, & Soboroff, 2007) found that opinion detection systems incorporating spam detection returned no or fewer spam documents in the top ten results than other systems. This is particularly attractive for applications such as Web search engines, where precision among top results is paramount. However, the overall performance of opinion finding was not affected, perhaps due to the low accuracy of spam detection in blogs.

[14] http://www.netlingo.com/

2.4.4 Implicit Opinion-Topic Association

The value of opinion detection has been found to increase when paired with the ability to determine the target topic of the opinion (Hurst & Nigam, 2004). However, it is often challenging to accurately identify the topic of an opinion: A document on a particular topic does not automatically suggest an association between the opinions it contains and the topic it is about. For instance, the opinion-bearing sentence "The seat in the theatre is very uncomfortable" in a movie review is not relevant to the movie. This may be solved, in part, by restricting the distance between the topic and the opinion. However, there are cases where the target topic occurs together with the opinion-bearing feature in a single sentence, yet the opinion is not about the target topic. For example, the sentence "After talking with my friend on Skype, I am depressed" cannot qualify as an opinion about "Skype." In this case, syntactic parsing may be able to resolve the problem by detecting no "modifying" relation between "depressed" and "Skype." Use of anaphors (i.e., pronouns that refer to previously mentioned people or things) makes targeted opinion detection even more challenging. One blog post may talk about the release of a new model of iPod and a subsequent comment may say "I love it!!!." This is obviously a positive opinion about the new iPod model, but it is hard for a machine to relate this opinion to the iPod, even with syntactic parsing.

There are two common strategies for tackling the challenge of implicit opinion-topic association. One strategy associates opinions and topics based on

proximity measured as the distance between opinion-bearing cues and topical keywords: Several of TREC's Blog track participants have found that opinion-bearing cues located close to topical words were good opinion indicators (Ding, Liu, & Yu, 2008; Vechtomova, 2007; Zhang et al., 2007; Zhou, Joshi, & Bayrak, 2007). The other strategy finds modifying relationships between topics and opinions using syntactic parsing and relationship analysis (Bloom et al., 2007; Nasukawa & Yi, 2003; Yi & Niblack, 2005). Although linguistic analysis appears to be more reliable than proximity based approaches for finding associations between opinions and topics, it carries additional computing expenses.

Although several researchers are aware of the potential benefits of resolving the problem of anaphora for improving recall in targeted opinion detection (Chklovski, 2006; Hurst & Nigam, 2004; Nasukawa & Yi, 2003; Nigam & Hurst, 2006), there have been no efforts to address this problem. This may be due to the complexity of the anaphora problem, which is difficult to solve even with deep linguistic analysis.

Although the challenges of noisy Web data and implicit opinion-topic association are crucial factors for opinion detection in the blogosphere and have hindered participants in TREC's Blog track from improving opinion retrieval performance, overcoming these challenges will likely depend on research efforts outside opinion detection: Web mining, spam detection, and anaphora resolution. However, the challenges of context sensitivity and domain dependency are

directly related to current problems in opinion detection, and strategies that address these latter challenges are of interest for further work in opinion detection.

2.4.5 Insufficiency of Labeled Data

Opinion-labeled data is essential for creating and evaluating a supervised classifier and for evaluating opinion-bearing features in the case of rule-based opinion detection. Extremely large amounts of labeled data are beneficial for acquiring a broad and comprehensive collection of opinion-bearing features (Riloff & Wiebe, 2003); for addressing the challenge of context sensitivity by providing informative high order n-grams; for sidestepping the challenge of domain dependency; and for allowing testing and evaluation of opinion detection strategies with high confidence. Despite the essential role of labeled data in implementing any opinion detection system, the reality is that labeled data collections in the target data domain are usually limited, especially at the sub-document level.

Opinion-labeled data can be prepared manually. Although manual labeling normally generates high-quality data, it is labor intensive in terms of designing, evaluating and following annotation rules. Manual labeling is therefore difficult to scale up. The Web service Amazon Mechanical Turk[15] allows researchers to hire large numbers of annotators without great cost; however, two important issues associated with this service are how to distribute data among annotators and how to control for annotation quality.

[15] https://www.mturk.com/mturk/welcome

Opinion-labeled data can be generated automatically by assuming opinion inheritance: that every sentence in an opinion-bearing document expresses an opinion and that every sentence in a factual document is factual. For example, Zhang and Yu (2007) constructed search queries by binding together an opinion target (e.g., "Skype") with patterns such as "I like" or "I don't think" and submitted it to a general search engine such as Google, whose top results were saved as positive training examples for opinion detection; they then searched the opinion target against Wikipedia to generate negative examples (i.e., non-opinion documents). This approach for preparing labeled data worked well for their opinion detection system.

While the opinion inheritance assumption may be acceptable in the case of review data, it is generally unreliable for other domains. Wiebe et al. (2001) examined a dataset consisting of articles from the *Wall Street Journal* and found that 30% of the sentences in opinion articles were objective, while 44% of the sentences in non-opinion articles were actually subjective. The proportion of opinion-bearing sentences found in opinion documents may actually be the same, if not higher, for blog posts.

While obtaining opinion-labeled Web data can be expensive and difficult, fetching unlabeled Web content in the target domain is normally cheap and easy. Unlabeled data have proven useful in identifying opinion-bearing features as well as in classifying opinion sentences. For example, augmenting a small amount of labeled data with large amounts of unlabeled data (e.g., the entire data collection)

can help identify informative opinion-bearing features such as unique words and unique n-grams (i.e., n-grams within which the unique word is replaced by a placeholder, UNIQ) (Dave et al., 2003; Wiebe et al., 2004). More often, unlabeled data are used in SSL, which does not focus on preparing more labeled data but on automatically and simultaneously learning and labeling the unlabeled data. In contrast with supervised learning, the value of SSL in opinion detection lies not only in its need for less labeled data but also in its ability to handle the domain dependency challenge: When there are labeled data in the non-target data domain only, an SSL algorithm can reduce the bias of the non-target data by increasing the number of labeled data from the target data domain. This aspect of SSL is very attractive for opinion detection in challenging data domains such as the blogosphere, which is short of high-quality opinion-labeled data.

3 Research Questions

Inspired by preliminary successes in the application of SSL for opinion detection and motivated by the challenges of insufficient labeled data in the blogosphere, the objective of this study is to investigate the value of SSL algorithms for improving the performance of opinion detection in the blogosphere. SSL has proven helpful in boosting the performance of topical text classification and in natural language processing tasks such as part-of-speech tagging (Søgaard, 2010) and word sense disambiguation (Y.Z. Niu, Ji, & Tan 2005; Yarowsky, 1995). Although SSL is especially attractive because it requires only a small number of labeled data, it has not been fully explored in opinion detection, and its potential value has not been investigated by comparing it to existing opinion detection approaches.

To assess the feasibility and effectiveness of SSL, this research compares the performance of a range of SSL algorithms with corresponding supervised learning-based algorithms to determine if an SSL-based opinion detector can significantly surpass a supervised opinion detector using the same amount of labeled data and if the best performance of the SSL-based opinion detector with less labeled data can approach or even outperform that of the supervised opinion detector.

There are many SSL algorithms, each based on different assumptions about the unlabeled data. Comparing the performance of different SSL algorithms in

opinion detection in the blogosphere contributes to a deeper understanding of opinion detection and its associated problems. Ideally, researchers choose the method(s) whose biases fit the target problem.

SSL algorithms can be divided into two general groups: those that employ only one classifier (e.g., self-training) and those that use multiple classifiers (e.g., co-training). The core assumption of co-training is that, by training one classifier with examples labeled by another classifier, the resulting union of classified examples will be better balanced than using either classifier alone. Previous SSL methods applied in opinion detection have used self-training (e.g., Riloff & Wiebe, 2003; Riloff et al., 2003). Some researchers have chosen not to apply co-training to opinion detection because they found it difficult to satisfy the requirement of the original co-training algorithm: That is, the set of opinion features should split into two parts, each of which is sufficient to make independent predictions for opinion class labels. However, other researchers have found that even random feature splitting has proven to yield better performance than self-training in the fields of topical text classification and NLP if there is sufficient redundancy in the feature set (Ng & Cardie, 2003; Nigram & Ghani, 2000). Motivated by the success of less-restrictive versions of co-training and enlightened by the amount and diversity of opinion-bearing features, this research examines the effectiveness of co-training for opinion detection by comparing it with self-training.

This research also examines the effectiveness of SSL algorithms when using opinion-labeled data from a non-target data domain. In TREC's Blog track, one

group used features extracted from non-blog Web data with a supervised opinion detector and achieved a maximum of approximately 10% improvement in overall performance over the baseline run in topical retrieval (Ounis et al., 2008). The current research assumes that SSL algorithms can better deal with the problem of domain dependency than supervised learning algorithms by gradually introducing targeted data and thus diminishing bias from the non-target dataset.

Specifically, this research is intended to answer the following questions:

Is SSL is effective for identifying opinions in Web content?

Can new SSL strategies be identified for use in opinion detection?

Can use of SSL strategies mitigate the problem of domain transfer?

Answering these three research questions will provide valuable guidelines and evaluation baselines for subsequent opinion detection studies using SSL algorithms, whether to integrate selected SSL algorithms within an existing opinion detection system or to build an SSL-based opinion detection system from scratch.

4 Methodology

4.1 Selection of Datasets

Because a document is normally a mixture of facts and opinions, sub-document level opinion detection is more useful and meaningful than document-level opinion detection. For this reason, all experiments were conducted on the sentence level.

Three types of text have been explored in prior opinion detection studies: news articles, online reviews, and online discourse in blogs or discussion forums. These texts differ from one another in terms of structure, text genre (e.g., level of formality), and the proportion of opinions each contains. A dataset of each type was selected in order to investigate the robustness and adaptability of SSL algorithms for opinion detection and to test the feasibility of SSL for domain adaptation. A small set of blog data was also used for parameter optimization. Several manually created opinion lexicons used in earlier studies were also collected in order to increase classification precision for data domains that are difficult for opinion detection.

One of the standard datasets in opinion detection is the movie review dataset created by Pang and Lee (2004)[16]. It contains 5,000 subjective sentences or snippets from the Rotten Tomatoes[17] pages and 5,000 objective sentences or

[16] This dataset can be downloaded from http://www.cs.cornell.edu/people/pabo/movie-review-data/, under subjectivity datasets.
[17] http://www.rottentomatoes.com/

snippets from IMDB[18] plot summaries, all in lowercase. Sentences containing less than 10 tokens were excluded, and the dataset was labeled automatically by assuming opinion inheritance.

The news article dataset[19] created by Wiebe et al. (1999) is widely used as the gold-standard corpus in opinion detection research. They chose the *Wall Street Journal* portion of the Penn Treebank III (Marcus, Santorini, Marcinkiewicz, & Taylor, 1999) and manually augmented it with opinion related annotations. According to their coding manual, subjective sentences are those expressing evaluations, opinions, emotions, and speculations. For this research, 5,297 subjective sentences and 5,174 objective sentences were selected based on the presence or absence of manually labeled subjective expressions.

The JDPA corpus[20] (Kessler, Eckert, Clark, & Nicolov, 2010), a new opinion corpus released in 2010, consists of blog posts expressing opinions about automobiles and digital cameras. Opinions about named entities (e.g., "seat", "lens") were manually annotated. All sentences containing sentiment-bearing expressions were extracted and objective sentences were manually identified by eliminating subjective sentences that were not targeted to any labeled entities. This process produced 10,000 subjective sentences and 4,348 objective sentences. To balance the number of subjective and objective sentences, 4,348 subjective sentences were randomly selected from the original set of 10,000.

[18] http://www.imdb.com/
[19] This dataset can be downloaded from http://www.cs.pitt.edu/mpqa/databaserelease/.
[20] The license form for this dataset is available at: http://www.icwsm.org/data/JDPA-Sentiment-Corpus-Licence-ver-2009-12-17.pdf

From 2006 through 2008, a dataset called Blogs06[21] was used for tasks in TREC's Blog track. Researchers at the University of Glasgow crawled the blogosphere over an 11-week period from December 2005 to February 2006 to create the Blogs06 collection (Ounis et al., 2007). In this collection, permalink documents (i.e., Web pages containing a single blog post with its associated comments) were the retrieval and assessment units. For TREC's Blog track opinion retrieval tasks, 50 topics (i.e., search queries and descriptions) were released every year, and each participant in the Blog track was to submit several retrieval runs, each run consisting of the top 1000 documents retrieved for each topic. The top documents retrieved across systems for each topic were then manually labeled as topical relevant, topical relevant but not opinion-bearing, and topical relevant and opinion-bearing (i.e., "positive," "negative," or "neutral"). Because topical relevance and opinion polarity would not be taken into consideration in this research, non-relevant data were ignored, and negative, positive and mixed opinion data were combined into one opinion dataset.

The Blogs06 collection is labeled at the document level and thus required manual labeling to prepare labeled data at the sentence level. In order to avoid bias caused by a particular topic, five TREC labeled opinion-bearing documents (1 positive, 1 negative and 3 mixed opinion) were randomly selected and manually examined for each of the 150 topics, for a total of 750 documents. Because machines cannot be expected to recognize trivial opinion expressions about which humans are uncertain, emphasis was placed on identifying opinion

[21] This dataset can be purchased via this page:
http://ir.dcs.gla.ac.uk/test_collections/access_to_data.html

expressions that contained explicit opinion cues. For example, in a product review, the sentence "I returned this product after a week" may indicate a negative opinion, but it may also state the fact that the product was returned because the reviewer received another as a gift. It is also reasonable to assume that explicit opinion cues may exist around ambiguous opinion expressions to support or explain them (e.g., "It is horrible! I returned this product after a week."). Therefore, a sentence was labeled as an opinion only if strong traces of opinion cues were present. Sentences that made objective statements were labeled as non-opinion, and the remaining sentences in selected blog posts were ignored. All in all, 1,237 subjective sentences and 616 objective sentences were collected.

4.2 Domain Independent Opinion Lexicons

Several studies have suggested that the use of high-quality opinion lexicons can yield high precision for opinion detection. Therefore, it is advisable to apply these lexicons to boost the classification precision of the initial classifier for SSL runs, especially for difficult data domains such as blog posts. Accordingly, six domain independent opinion lexicons that had proven useful in previous opinion mining studies were collected for use in these experiments.

Adjectives are often connected to the expression of attitudes and have been reported to have a positive and statistically significant correlation with subjectivity (Wiebe et al., 1999). Three types of adjectives—dynamic adjectives, gradable adjectives, and semantic oriented adjectives—have been found to be stronger subjective cues than adjectives as a whole (Hatzivassiloglou & Wiebe, 2000). *Dynamic adjectives* are adjectives with "qualities that are thought to be

subject to control by the possessor and hence can be restricted temporally" (Quirk, Greenbaum, Leech, & Svartvik, 1985, p. 434). For example, a dynamic adjective such as "foolish" can be used in the sentence "he is being foolish" while non-dynamic (i.e., stative) adjective such as "tall" is not appropriate in the sentence "he is being tall." Bruce and Wiebe (1999) proposed that the stative/dynamic distinction between adjectives was related to subjectivity and manually identified a list of 120 or so dynamic adjectives in roughly 500 sentences from the *Wall Street Journal* Treebank Corpus. Preliminary examination indicated that these dynamic adjectives were more subjective than other adjectives in the corpus.

Using a different corpus, Hatzivassiloglou and Wiebe (2000) confirmed the strong correlation between dynamic adjectives and subjectivity in the news domain with more than 30% improvement in precision over adjectives as a whole. They suggested that a second type of adjectives, which they called *gradable adjectives*, were also useful indicators of opinions and demonstrated 13-21% higher precision than adjectives as a whole. Gradable adjectives are those which can participate in comparative constructions (e.g., "This movie is *more* exciting than the other") and can accept modifying expressions that act as intensifiers or diminishers (e.g., "This game is *very* exciting", where "very" is an intensive modifier) (Hatzivassiloglou & Wiebe, 2000).

The third type of adjective is a more intuitive form of opinion evidence. Semantic oriented adjectives are polar words that are either positive or negative. Adjectives with polarity, such as "good", "bad", or "beautiful", are inherently connected with opinions as opposed to adjectives, such as "white", "Chinese", or

"skinny", which have no polarity. Hatzivassiloglou and Wiebe (2000) and Whitelaw, Garg, and Argamon (2005a, 2005b) demonstrated that a reasonably high accuracy of either opinion detection or polarity detection could be achieved by using polar adjectives alone. Chesley et al. (2006) also found that positive adjectives played a major role in classifying opinionated blog posts.

Three adjective opinion lexicons were selected for this research: Index of General Inquire (IGI) tag categories, a manually constructed list that contains 765 positive and 873 negative words (Stone, 1997); Colin adjectives, an opinion lexicons distributed by Hatzivassiloglou and Wiebe (2000), which include manually and automatically identified semantic oriented adjectives, dynamic adjectives and gradable adjectives; and strong semantic oriented adjectives in the subjectivity term list created by Wilson et al. (2003). Dynamic adjectives were separated from other Colin adjectives into an individual lexicon because of their unique feature and their significant contribution.

Although not as significant as adjectives, verbs have also been found to be good indicators of opinion information. *Verb classes*, categories for classifying verbs syntactically and/or semantically, are often used for culling opinionated verbs. *Levin's verb classes,* developed on the basis of both intuitive semantic groupings and participation in valence, or polarity, alternations (Levin, 1993), are the most popular verb classes used as opinion evidence. There are 193 Levin's verb classes, which are grouped into 51 sections with two additional levels of subsections. For this research, verbs from opinion-related Levin's verb classes, including *judgment* (e.g., "abuse," "acclaim"), *complain* (e.g., "hate," "despise"),

and *psych* (e.g., "amuse," "admire," "marvel (at)"), were selected. Similarly, *FrameNet* (Fillmore & Baker, 2001), which groups words, including verbs, according to conceptual structures, provides semantic frames such as *communication* (e.g., "indicate", "convey") as evidence of opinion (Breck, Choi, & Cardie, 2007). For this research, several frames were selected: "agree or refuse to act", "be in agreement on assessment", "desirability", "experiencer(objective/subjective)", "judgment", "opinion", "prevarication" and "statement".

Another linguistic form providing evidence of opinion is the *appraisal group* (Whitelaw et al., 2005a, 2005b). An appraisal group is a sophisticated linguistic feature at the phrase level that is comprised of a head term with a defined attitude type and an optional list of preceding appraisal modifiers. For example, "not very happy" is an adjective appraisal group with "happy" as the head term and "not" and "very" as modifiers. Appraisal groups have also been suggested as useful in identifying what is called an *appraisal expression*, "a textual unit expressing an evaluative stance towards some target" (Bloom et al., 2007, p. 308). Given the high cost of full syntactic parsing and the difficulty of fine-level analysis, this research used only the head adjectives, which are marked as positive or negative in the hand-built lexicon distributed by Bloom et al. (2007).

In addition to single words, opinion lexicons used in this research include patterns such as IU collocations (Yang et al., 2007) and bigrams. IU collocations are n-grams with first-person pronouns (e.g., "I", "we") and second-person pronouns (e.g., "you") as anchor terms. During their experiments for TREC's

Blog track, Yang et al. (2007) found that IU collocations worked best as single features. The UMass Amherst Linguistics Sentiment Corpora (Constant, Davis, Potts, & Schwarz, 2009; Potts & Schwarz, 2008) consists of unigrams and bigrams gathered from online book reviews on Amazon[22] and online hotel reviews on TripAdvisor[23]. For each n-gram, total occurrence is reported on an ordinal scale of 1 to 5, with 1 indicating a highly negative review and 5 indicating a highly positive review. In order to pick opinion n-grams, bigrams were excluded if they: contained domain stop words (e.g., book, hotel); occurred frequently at all rating levels; occurred more often at neutral ratings than at either positive or negative ratings; or contained digits or less than 3 characters. Only those n-grams appearing in both Amazon book reviews and TripAdvisor hotel reviews were retained.

Altogether, nine domain-independent opinion lexicons were utilized: appraisal semantic oriented adjectives[24], gradable and semantic oriented Colin adjectives, dynamic adjectives[25], IGI semantic oriented adjectives[26], Wilson subjective terms[27], Levin's opinion-related verb class terms (see Appendix A), FrameNet opinion related category labels (see Appendix B), IU collocations (see Appendix C), and review bigrams (see Appendix D).

---

[22] http://www.amazon.com/
[23] http://www.tripadvisor.com/
[24] The appraisal adjectives can be downloaded from http://lingcog.iit.edu/arc/appraisal_lexicon_2007a.tar.gz
[25] The gradable and semantic oriented Colin adjectives and the dynamic adjectives can be downloaded from http://www.cs.pitt.edu/~wiebe/pubs/coling00/coling00adjs.tar.gz
[26] The IGI words can be accessed at http://www.wjh.harvard.edu/~inquirer/inqdict.txt. Positive and negative words were extracted.
[27] The Wilson subjective terms are included in the OpinionFinder package available at http://www.cs.pitt.edu/mpqa/opinionfinderrelease/. Strong subjective terms were extracted.

4.3 Data Preprocessing

All words in datasets were converted to lower case, and numbers were replaced with the placeholder "#". Unigrams and bigrams were generated for each sentence, and common stop words such as articles and prepositions (see Appendix E) were removed from unigrams. For movie review data, words such as 'movie' and 'film' were also removed because they are frequently used in both movie reviews and plot summaries and are thus not good indicators of an opinion or non-opinion class. No domain specific stop words were removed from the other two data domains because there are no specific topics associated with the news article dataset and, although the JDPA blog dataset contains car and camera reviews, the presence of words such as 'camera' or 'car' may be an indicator for an opinion or non-opinion class. No stemming was conducted since the literature shows no clear gain from stemming in opinion detection; stemming may actually erase subtle opinion cues such as past tense verbs. For each sentence, nine lexicon scores were assigned, with each score corresponding to the total occurrence of a term in one particular lexicon.

As illustrated in Figure 6, each dataset was randomly split into three portions: 5% of the sentences were reserved as the evaluation set (E) and were available only for S^3VM runs; 90% were treated as unlabeled data (U); and i% (i = 1, 2, 3, 4 or 5) were treated as labeled data (L).

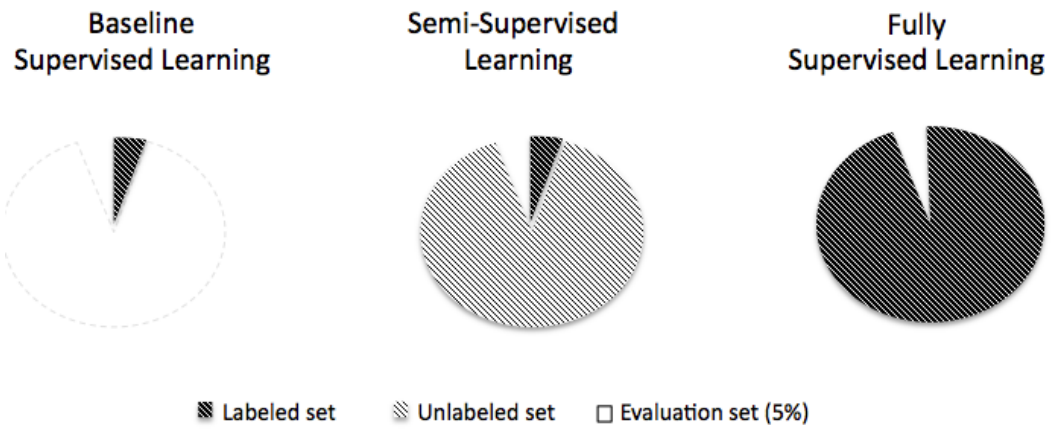

Figure 6. Data split for semi-supervised learning runs, baseline supervised learning runs and fully supervised learning runs.

4.4 Experimental Design

In the experiments reported here, opinion detection was treated as a binary classification problem with two categories: subjective sentences (i.e., positive examples, or *p*) and objective sentences (i.e., negative examples, or *n*). In all experiments, evaluation of performance was based on classification accuracy. Although four major semi-supervised learning (SSL) methods—self-training, co-training, EM-NB, and S^3VM—were investigated, the research focused on self-training and co-training because both are wrapper approaches that can be easily adopted by any existing opinion detection system. Three groups of experiments were conducted to investigate the effectiveness of SSL approaches in opinion detection: The first group of experiments was designed to asses SSL methods using one classifier; the second group was designed to develop and evaluate co-training strategies; and the third group was designed to examine the applicability of SSL for domain adaptation.

4.4.1 Design of Experiment 1: SSL with One Classifier

This group of experiments investigated SSL methods using a single classifier, whether self-training, EM-NB or S^3VMs.

4.4.1.1 Baseline Runs

SSL algorithms are expected to perform better than supervised learning algorithms (i.e., baseline supervised algorithms) using the same number of labeled data since the former hold more knowledge than the latter. According to the literature of SSL, any advantage of SSL will degrade with increasing numbers of labeled data (Nigam & Ghani, 2000). Furthermore, supervised learning algorithms using all labeled data (i.e., fully supervised learning algorithms) are expected to perform better than an SSL algorithm using the same number of partially labeled and partially unlabeled data.

If SSL algorithms are effective, they should significantly outperform the baseline for supervised learning algorithms and should approach the performance of fully supervised learning algorithms. Therefore, in order to test the effectiveness of SSL algorithms with respect to the number of available labeled data, each SSL opinion detector was trained on the labeled dataset *L* and the unlabeled dataset *U*. The corresponding baseline supervised opinion detector was constructed using only *L*, and the fully supervised opinion detector was constructed by treating all data in *U* and *L* as labeled data. Performance of each

SSL run was compared with the performance of both the baseline SL run and the full SL run.

4.4.1.2 Classification Algorithm and Feature Representation

Although both Naïve Bayes and SVM are the most effective text classification algorithms and have been commonly employed in opinion detection systems, only the Naïve Bayes classifier was selected for self-training because preliminary experiments showed that, even with a logistic model to output probability scores for the SVM classifier, the difference in probabilities is too small to select a small number of top classification predictions.

Opinion detection usually requires more features than single words to capture subtle opinions. Therefore, for each sentence, both unigrams and bigrams were extracted as classification features. Higher order n-grams (i.e., n>=3) were not used because effective high order n-grams cannot be extracted from a small labeled dataset. Features with binary values (i.e., presence or absence) were used, motivated by the brevity of the text unit at the sentence level as well as by the characteristics of opinion detection, where frequency of occurrence has proven to be less influential.

4.4.1.3 Parameter Settings

Table 2 shows three SSL algorithms and the corresponding supervised learning algorithms to which they were compared. The Naïve Bayes classifier was used as the base classifier for both self-training and EM-based SSL. SVM was

compared with its SSL counterpart S^3VMs. Other parameter settings included: (1) for all SSL algorithms, iterations stopped when there was no more unlabeled data; (2) for each iteration, a number of unlabeled examples *u*, smaller than *U*, were randomly extracted from the unlabeled dataset *U* for classifiers to predict opinion labels; (3) for each iteration, opinion examples (*p*) and non-opinion examples (*n*) were added back to the labeled dataset. The ratio between *p* and *n* approximates the distribution of opinions and non-opinions in the labeled dataset; and (4) for the EM-based Naïve Bayes classifier (i.e., EM-NB) and S^3VMs, the default parameter settings in LingPipe (Alias-i, 2008) and SVM^{light} (Joachims, 1999a) were adopted, respectively.

Table 2

General Experiment Design for SSL

SSL algorithms	SSL parameters	SL	Initial labeled data
Self-training	*u, p, n*	Naïve Bayes	1, 2, 3, 4 and 5 % of all labeled data (apply to all)
EM-based	default LingPipe	Naïve Bayes	
S^3VMs	default SVM^{light}	SVM	

Note. SL = supervised learning; *p* = # of positive examples; and *n* = # of negative examples.

4.4.2 Design of Experiment 2: Co-Training Strategies

The second group of experiments was designed to examine whether there are co-training strategies for opinion detection that can reduce the bias caused by a particular opinion detector if there is no nature split in the feature space. Different

strategies for co-training were investigated with an emphasis on strategies for creating two classifiers with different views.

Five strategies were applied to generate view classifiers: (1) using unigrams and bigrams respectively to create two view classifiers based on the assumption that there is enough redundancy in low order n-grams and high order n-grams. In this investigation, unigrams were understood as content features and higher order n-grams as context features. Bigrams were actually used in combination with unigrams because bigrams alone are weak features when extracted from limited labeled data at the sentence level; (2) randomly splitting a feature set in two to train two view classifiers (Nigram & Ghani, 2000); (3) creating two view classifiers by using two different supervised learning algorithms (e.g., Naïve Bayes and a rule-based classifier) under the assumption that they will provide useful examples to each other because they are based on different learning assumptions; (4) training two view classifiers based on a random split of the training set; and (5) applying a character-based language model (CLM) and a bag-of-words model (BOW) since the former takes into consideration the sequence of words while the latter does not.

During every iteration, auto-labeled sentences were selected to expand the original labeled dataset if both classifiers agreed on the assigned label with highest confidence. Forcing agreement on confident predictions helped to establish and maintain a relatively high level of classification precision, especially when initial classifiers were not effective alone.

The baseline supervised run for co-training used two classifiers with a simple voting strategy: For each new example, the classifier with the higher prediction score decided the final class prediction. The performance of each co-training run was also compared with the performance of its corresponding self-training algorithm. For example, co-training runs using two different Naïve Bayes classifiers were compared with self-training runs using each of the Naïve Bayes classifiers. Parameters *k, u, p, n* were set at the optimized values established in the first group of experiments.

4.4.3 Design of Experiment 3: Domain Adaptation

This group of experiments used opinion-labeled data from one or more non-target data domains to examine possible solutions for opinion detection in challenging data domains. Because the movie review data achieved the greatest percentage of classification accuracy in preliminary testing, they were treated as the source data, while datasets for news articles and blog posts were treated as target data.

While the data split for the target domain was the same as that used in previous experiments, all sentences in the source domain, except for the 5% evaluation data, were treated as labeled data. For example, in order to identify opinion-bearing sentences from the blog dataset, all 9,500 movie review sentences and i% of blog sentences were used as labeled data, 90% of blog sentences were used as unlabeled data, and 5% were reserved as evaluation data. In addition, a parameter was added to gradually reduce the weight of non-blog examples in the

training set during iterations, similar to the approach taken by Tan et al. (2009). To reduce bias caused by features specific to one non-target data domain, labeled data from two different non-target data domains were combined as training data for both supervised and semi-supervised learning algorithms (i.e., in co-training, two view classifiers were trained on two non-target domains).

In order to compare the benefits of employing non-target labeled data to the benefits of using general opinion lexicons to deal with the domain transfer problem, another set of domain adaptation experiments used general opinion lexicons instead of borrowing opinion labeled sentences from other domains. In addition to the n-gram features, SL and SSL runs in this set used features from nine opinion lexicons to represent each in-domain sentence.

4.5 Evaluation Measures

Opinion detection was treated as a classification task in this research, and classification accuracy was used as the evaluation measure when comparing SSL and SL runs. Classification accuracy evaluates the overall correctness of a classifier and is calculated using the formula ACC = (a+d)/(a+b+c+d).

To compute classification accuracy, a contingency table was generated to present easy-to-read classification results and to provide basic statistics for more complicated measurements. For example, in Table 3, *a* and *d* represent the number of examples where the classifier made the correct decision, while *b* and *c* represent the number of examples where the classifier made an incorrect decision.

Table 3

Contingency Table for Opinion Detection

	Gold Standard	
Classifier Prediction	Opinion	Non-opinion
Opinion	a	b
Non-opinion	c	d

Since the motivation for applying SSL was the value of unlabeled data for augmenting labeled data, there are other measures that can be used specifically for evaluating the value of additional unlabeled data. Abney (2008) summarized two such measures: (1) optimized performance when augmenting a fixed number of labeled data with a growing number of unlabeled data without bound; and (2) the relative value to human efforts of labeling data by comparing the amount of additional unlabeled data and labeled data necessary to achieve the same performance. These two measures were adopted to determine whether performance increased when more unlabeled data were used and whether the contribution of unlabeled data decreased with the increase in available labeled data, as suggested in most SSL studies (e.g., Nigam & Ghani, 2000).

5 Results and Discussion

This chapter reports on the performance of semi-supervised learning (SSL) methods using one classifier only, the performance of various co-training strategies, and the performance of self-training and co-training for handling the domain transfer problem. The results of all SSL runs are compared with the corresponding baseline and full SL results.

5.1 Preliminary Experiments

Self-training runs with various parameter settings were conducted on TREC's blog data to evaluate the impact of different experimental settings and to determine optimized parameters for all self-training and co-training runs.

5.1.1 Feature Selection

Feature selection is an important practice in topical text classification, where it is used to reduce the dimensionality of the feature space by selecting salient features according to certain feature-category similarity scores. Feature selection can reduce computational costs as well as potential overfitting, which is normally caused by training on data that is either biased or too specialized. Previous studies have used feature selection successfully to determine opinion-bearing features (Abbasi et al., 2008; Gamon, 2004; Ng et al., 2006; Yi et al., 2003).

Two popular feature selection methods—information gain (IG) and chi-square (CHI)—were investigated for SSL. The IG value of a feature *F* with respect to a class *C* is the reduction in uncertainty about *C* by knowing *F*, where the uncertainty of the class *C* is measured by its entropy. Chi-square measures the lack of independence between a feature *F* and a class *C*, using the chi-square distribution for extremeness judgments. When keeping all other parameters fixed and selecting the top 100 features, neither feature selection method contributes to SSL performance with labeled data from 1% to 5% of the total dataset. Because feature selection consumes computing time, especially when a new classification model must be built for each iteration, no feature selection was conducted for the subsequent experiments.

5.1.2 Unlabeled Data Available for Each Iteration

To decide how many unlabeled sentences *u* should be available to the classifier on each iteration, experiments were designed using 20, 75, 100 and all unlabeled sentences (i.e., approximately 1660 sentences). By computing the average improvement of self-training runs over corresponding baseline SL runs with 1% to 5% labeled data, it was found that self-training runs classifying all unlabeled sentences on each iteration decreased classification accuracy by 4.67%; self-training runs classifying 100 unlabeled sentences on each iteration increased baseline performance by 2.08%; self-training runs classifying 75 unlabeled sentences on each iteration did not improve baseline performance; and self-training runs classifying only 20 unlabeled sentences on each iteration increased baseline performance by 4.18%. For the following experiments, *u* was set to 20.

After p auto-labeled opinion sentences and n auto-labeled non-opinion sentences were selected and added to the labeled dataset, p+n unlabeled sentences can be drawn from U to replenish u or a new set of u can be generated from U. Experiments using TREC's blog dataset indicated that replenishing u outperformed generating a new set of u by 11.87% in terms of classification accuracy. One explanation is that, for succeeding iterations, replenishing u kept those unlabeled sentences for which the classifier generated low prediction scores in the current iteration and forced the classifier to reclassify difficult sentences, while generating a new set of u allowed the classifier to select sentences that were easy to classify.

On the one hand, in order to avoid mislabeled data in the labeled dataset, only the most confidently labeled data should be selected, and a small value for p and n would be preferred. On the other hand, in order to reduce the number of iterations necessary for SSL to converge, a larger value for p and n would be preferred. Preliminary experiments compared the results of setting p and n either to one or to two and found no noticeable difference. For this reason, p and n were set at two for all experiments.

5.2 SSL Using One Classifier

The first experiment examined the effectiveness of SSL methods that required only one classifier: self-training, EM-NB and S^3VM. For the movie review, news and blog data domains, the performance of three SSL runs were

compared with the performance of SL runs that used the same number of labeled sentences as well as those that used all data as labeled sentences.

Table 4

Classification Accuracy (%) of Self-Training and Supervised Learning Runs for Movie Reviews

	# of Original Labeled Sentences				
Run Type	100	200	300	400	500
Baseline SL	63.80	73.60	77.20	79.40	80.20
Self-training	85.20	86.60	87.00	87.20	85.20
EM-NB	88.10	88.70	88.60	88.40	89.00
S^3VM	69.60	74.00	75.00	76.80	80.40
Full SL	90.00	92.00	91.80	91.60	91.80

Note. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9000 labeled sentences.

Table 4 shows the classification accuracy of three SSL methods and two supervised learning runs for movie reviews. Self-training and EM-NB always outperformed the corresponding SL baseline on movie reviews: At convergence, SSL methods achieved improvement in the range of 8% to 34% over the SL baseline. The fewer the initial labeled data, the more benefit an SSL run gained from using unlabeled data.

The performance of baseline supervised learning runs using 100 to 500 labeled sentences is shown in the first row of Table 4. The more labeled data provided for the baseline SL runs, the better the performance: With 100 labeled

sentences, the baseline SL run achieved classification accuracy of only 63.80%; but, with 500 labeled sentences, the supervised learning classifier achieved classification accuracy of 80.20%. The second row shows the performance of the simple self-training method using 100 to 500 labeled sentences and an additional 9000 unlabeled sentences. These self-training runs improved performance over the corresponding baseline supervised runs: For example, using 100 labeled sentences, self-training achieved a classification accuracy of 85.2% and outperformed the baseline SL by 33.5%. Although the full SL run using all labeled data surpassed the simple self-training run by 4.9%, significant effort was saved by labeling only 100 sentences rather than 9,100. With 500 labeled sentences, self-training improved accuracy over the baseline supervised run by 6%, indicating that self-training is particularly beneficial when the number of labeled data is small.

EM-NB is similar to self-training but optimizes classification parameters on each iteration. It showed greater improvement over the supervised baseline than simple self-training: 38% to 11% increases over baseline runs were achieved by EM-NB runs. Furthermore, the gap between EM-NB runs and the fully supervised runs was as small as two to three percentage points in terms of absolute value of classification accuracy.

S^3VM had the worst performance of the three SSL methods. When there were 100, 200 and 500 original labeled sentences, S^3VM showed only slight improvement over the supervised learning baseline; for all other runs, it actually hurt the performance. This indicates that the boundaries between opinion and non-

opinion classes in a vector space containing both labeled and unlabeled examples are not more dominant than those associated with other factors, such as topics, and are captured by a simple implementation of S^3VM without advanced parameter tuning.

Overall, SSL methods that iteratively labeled unlabeled data with one classifier were effective for movie reviews and achieved performance close to fully supervised learning while saving the labor involved in labeling thousands of unlabeled sentences.

Table 5

Classification Accuracy (%) of Self-Training and Supervised Learning Runs for News Articles

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL	60.50	64.31	69.47	69.47	71.38
Self-training	60.11	65.84	66.41	67.75	67.56
EM-NB	68.70	69.80	69.50	70.00	70.30
S^3VM	61.54	62.12	61.35	67.88	67.88
Full SL	77.29	76.72	76.91	75.95	75.76

Note. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9270 labeled sentences.

Table 5 reports the results of SSL runs using one classifier for the domain of news articles. Again, the more labeled data, the better the performance of

supervised learning runs; however, the classification accuracy of supervised learning runs was lower for news articles than for movie reviews. The initial supervised classifier with low accuracy produced low accuracy in the auto-labeled sentences. As a result, self-training, which used auto-labeled data to update the classifier during each iteration, did not improve the performance of the baseline supervised learning. EM-NB, which not only automatically labeled sentences but also updated prediction scores for each sentence on each iteration, showed slight improvement over the performance of baseline runs by a maximum of 14% for the run with 103 initial labeled sentences. S^3VM did not improve performance in four of the five runs.

Table 6

Classification Accuracy (%) of Self-Training and Supervised Learning Runs for Blog Posts

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	55.05	58.95	61.93	64.69	66.06
Self-training	54.59	55.73	56.65	58.49	64.45
EM-NB	55.40	57.90	58.90	59.80	59.90
S^3VM	55.02	49.40	55.42	60.04	60.64
Full SL	71.56	73.17	72.71	72.94	72.48

Note. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 7740 labeled sentences.

As shown in Table 6, none of the SSL runs proved beneficial in the blog domain. This is because the blog data domain is even more challenging than the news domain. The language used in blog posts is more informal than the language of the other two data domains, and blog writing contains a variety of opinion cues not found in movie reviews or news writing. Furthermore, because the JDPA blog data are focused on reviews of cars and cameras, opinion and non-opinion sentences share topic-related features; and the average length for opinion and non-opinion sentences in blog posts is 17 words, shorter than that for movie reviews (23.5 words) or news articles (22.5 words). In fact, approximately one quarter of the sentences in the blog dataset had only 5 to 10 words. This poses an additional challenge because there is less information for the classifier in terms of the number of individual features.

Table 7

Average Sentence Length in Different Data Domains

Data Domain	Average # of Words		
	Opinion Sentence	Non-Opinion Sentence	Overall
Movie Reviews	22	25	23.5
News Articles	25	20	22.5
Blog Posts	19	15	17

With limited labeled data, the results of these experiments suggest that both self-training and EM-NB methods can make effective use of unlabeled data for opinion detection in certain data domains (e.g., movie reviews) but not in others

(e.g., news and blog data). EM-NB produced slightly better performance than self-training, while S^3VM did not show any benefit across the three data domains, possibly because of the bias of unlabeled data.

5.3 Comparison of Co-Training Strategies

In order to overcome the possible bias of one particular classification algorithm and to make more effective use of limited labeled data, a series of SSL experiments applying different co-training strategies was conducted. The first set of SSL experiments investigated simple co-training configurations using different feature sets or different training sets. Because of the difficulty of finding a supervised classifier that can produce easily distinguishable prediction scores and the fact that the complicated supervised classification algorithm was not the focus of this research, adding another classifier for co-training was not included in these experiments. The performance of co-training runs using a character-based language model (character 8-grams) to train one classifier and a bag-of-words model to train the other classifier was also investigated using two classifiers that differed both in feature representation (i.e., character vs. word) and in the learning algorithm (i.e., language model vs. pure statistical model). In addition, other co-training strategies using simple linguistic features were also investigated.

5.3.1 Random Labeled Data Split

In the movie review domain, co-training runs using two classifiers trained with randomly split labeled data improved the performance of their corresponding baseline supervised runs and approached the performance of fully supervised runs.

Table 8

Classification Accuracy (%) of Co-Training with Random Labeled Data Split for Movie Reviews

	# of Original Labeled Sentences				
Run Type	100	200	300	400	500
Baseline SL	66.60	66.60	77.80	80.20	82.40
Co-training (converged)	83.20	88.60	87.00	87.80	87.00
Co-training (best)	85.00	89.60	87.40	88.20	88.60
Full SL	85.80	85.80	88.80	88.60	88.60

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9000 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 8 reports the performance of co-training and supervised runs using from 100 to 500 labeled sentences. The supervised learning results reported in Table 8 differ from the results for movie reviews reported in Table 4 because two classifiers were used to ensure a fair evaluation of the benefits of co-training. More specifically, the baseline supervised run for co-training run combined two classifiers, each of which was trained on half of the labeled data, and made final class predictions based on the classifier with the higher prediction score. Baseline

SL runs used 100 to 500 labeled sentences only; full SL runs used 9100 to 9500 labeled sentences.

The results of co-training using 100 to 500 initial labeled data are reported for converged performance (i.e., the performance of the last iteration) and for the best performance across all iterations during co-training. For example, the baseline supervised run using two classifiers, each trained with 50 initial labeled sentences, had a classification accuracy of 66.60%; co-training using the same two classifiers achieved an accuracy of 83.20% when performance converged and achieved a best performance of 85.00%, outperforming the baseline supervised run by more than 18% in absolute value; the full supervised run, which trained each classifier with 4550 initial labeled sentences, achieved a classification accuracy of 85.80%, gaining only 2.6% in classification accuracy over the converged co-training run but requiring 9000 additional labeled sentences. These results indicate that, the more labeled data that are available, the less helpful are unlabeled data and the smaller is the increase that co-training produces over baseline supervised learning.

Table 9

Classification Accuracy (%) of Co-Training with Random Labeled Data Split for News Articles

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL	61.07	64.70	67.18	66.80	66.99
Co-training (converged)	64.89	65.84	66.79	68.13	66.41
Co-training (best)	66.60	66.22	71.57	69.28	68.90
Full SL	74.81	75.57	74.62	75.19	75.00

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9270 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

In the news domain, co-training using split labeled data was able to improve the performance of baseline supervised learning slightly but only when the available labeled data were limited (i.e., 103 and 206 initial labeled sentences as indicated in Table 9). When there were 300 or more labeled sentences, co-training showed no benefit. This is due to the difficulty of the news domain as demonstrated by the fact that the full SL run using more than 9500 labeled sentences was only able to achieve a classification accuracy of approximately 75%.

Table 10

Classification Accuracy (%) of Co-Training with Random Labeled Data Split for Blog Posts

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	56.40	54.60	56.20	60.40	63.20
Co-training (converged)	52.80	52.20	52.20	53.20	57.20
Co-training (best)	57.40	57.60	58.80	61.80	64.80
Full SL	70.80	70.40	70.20	71.00	70.00

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 7740 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

As shown in Table 10, the co-training strategy decreased the performance of baseline supervised learning when applied to blog data. The finding that unlabeled data were not helpful can be explained by the same reasons provided for self-training classifiers dealing with blog data: informal and creative language used in blog writing as well as narrow topics and short sentence length of the JDPA blog data.

To understand the impact of poor baseline performance on the quality of auto-labeled data, mislabeled auto-labeled sentences were examined. For example, the co-training run using 86 labeled sentences mislabeled half of the auto-labeled sentences selected to replenish the labeled dataset on the first iteration, and the rate of mislabeled sentences remained constant around 37.5% for the first 10 iterations. Interestingly, for the top 50 opinion features of the converted

classification model, camera-related nouns such as “camera”, “canon”, and “lens” appeared to be more related to opinion expressions than non-opinion expressions, while car-related nouns often demonstrated the opposite phenomenon (i.e., car-related nouns appeared to be more related to non-opinion expressions than to opinion expressions). A closer look at the blog posts revealed that, when people wrote posts about cars, they tended to use personal pronouns (e.g., “she”) to refer to the car and sometimes even personified it (e.g., “I want her to last much longer than that (which is why I get oil changes, duh)”). This is not the case when they write about cameras, suggesting that personal pronouns can be valuable opinion indicators when they refer to the target topic/object. Although pronouns were treated as stop words and were removed during data pre-processing, this suggests that, in future research, anaphora resolution should be handled before removing stop words, even though the anaphora problem is complex and difficult to solve, even with deep linguistic analysis.

5.3.2 Random Feature Split

Table 11

Classification Accuracy (%) of Co-Training with Random Feature Split for Movie Reviews

	# of Original Labeled Sentences				
Run Type	100	200	300	400	500
Baseline SL	67.07	76.03	78.73	82.13	84.58
Co-training (converged)	82.20	86.80	88.50	87.29	88.60
Co-training (best)	84.57	87.97	89.48	88.52	90.00
Full SL	84.80	89.60	89.58	89.82	89.55

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9000 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 12

Classification Accuracy (%) of Co-Training with Random Feature Split for News Articles

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL	62.45	65.33	68.04	66.73	63.60
Co-training (converged)	67.67	69.52	71.03	69.22	69.47
Co-training (best)	69.88	71.94	72.47	71.03	73.77
Full SL	75.64	74.90	76.79	76.35	75.75

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9270 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 13

Classification Accuracy (%) of Co-Training with Random Feature Split for Blog Posts

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	54.36	53.99	57.03	59.22	63.03
Co-training (converged)	51.62	53.83	54.03	55.80	58.50
Co-training (best)	57.81	57.49	59.18	61.92	63.33
Full SL	71.26	71.14	70.65	70.77	71.08

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 7740 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 11, Table 12 and Table 13 report the performance of the co-training strategy using randomly split feature sets for movie reviews, news articles and the blog domain, respectively. The results show trends similar to those of the co-training strategy using a randomly split training set: Co-training showed improvement over the baseline supervised runs for both the movie review and news domains, approaching the results of full SL runs; but, co-training using a randomly split training set did not improve classification accuracy for the blog domain.

5.3.3 Unigrams and Unigrams plus Bigrams

Random split of either labeled data or features showed only limited benefits as co-training strategies for opinion detection. Although simple to implement, the random split strategy is not capable of capturing different views of the data and

thus does not take full advantage of co-training, which is most effective using two very different classifiers. Yet another co-training strategy uses word bigrams and unigrams to capture both the content and the context of target sentences.

Table 14

Classification Accuracy (%) of Co-Training for Movie Reviews Where One Classifier Uses Unigrams and the Other Uses Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	100	200	300	400	500
Baseline SL	67.60	75.20	80.20	81.80	83.80
Co-training (converged)	85.00	87.20	88.00	88.80	88.20
Co-training (best)	86.40	88.80	88.60	89.80	89.40
Full SL	86.80	90.40	90.40	90.40	90.40

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9000 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

For movie reviews, co-training using unigrams and bigrams improved baseline supervised learning by 26% with 100 labeled sentences and by 5% with 500 labeled sentences. The absolute classification accuracies of the co-training strategy using this n-gram-based feature split were higher than that of the co-training strategy using random feature split when the quantity of labeled sentences was small (e.g., 100) because the randomly split feature set reduced the features available for each classifier, thus decreasing performance when the total number of features was already limited. When co-training performance is compared to self-training using one classifier (see Table 4), a slight gain is achieved by introducing a second classifier. It appears that the more labeled data that is

available, the higher the quality of each classifier, the better the performance of co-training, and the greater the improvement of using two classifiers over one.

Table 15

Classification Accuracy (%) of Co-Training for News Articles Where One Classifier Uses Unigrams and the Other Uses Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL	60.69	64.31	69.28	66.79	66.41
Co-training (converged)	61.64	71.38	69.28	69.85	69.66
Co-training (best)	65.08	72.52	70.99	70.61	72.14
Full SL	74.81	76.15	75.95	75.95	75.38

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9270 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 16

Classification Accuracy (%) of Co-Training for Blog Posts Where One Classifier Uses Unigrams and the Other Uses Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	55.00	54.80	56.60	59.60	61.60
Co-training (converged)	50.20	52.80	53.60	54.40	57.80
Co-training (best)	57.80	55.80	58.80	61.20	65.20
Full SL	72.00	72.20	72.40	71.20	71.60

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 7740 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

For the news domain (see Table 15), co-training improved baseline supervised learning slightly; but even the best classification accuracies achieved across iterations were outperformed by full SL runs by 4% to 15%. For blog posts (see Table 16), the performance of co-training runs was actually lower than the performance of the baseline supervised learning runs, indicating that using unlabeled data in co-training for opinion detection did not show any benefit for the blog dataset.

5.3.4 Character-Based Language Model (CLM) and Bag-Of-Words Model (BOW)

Co-training strategies investigated so far used two classifiers trained with different subsets of word n-grams features. To obtain a more distinct pairing of classifiers, another co-training strategy used two classifiers that differed from each other not only in terms of feature representation but also in classification model. This co-training strategy applied a character-based language model (CLM) and a bag-of-words model (BOW), where the former takes into consideration the sequence of characters while the latter does not consider the sequence of word occurrences.

Table 17

Classification Accuracy (%) of Co-Training for Movie Reviews Where One Classifier Uses CLM with Character 8-grams and the Other Uses BOW Model with Word Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	100	200	300	400	500
Baseline SL	75.80	80.80	82.60	85.20	84.80
Co-training (converged)	92.20	93.80	92.60	93.20	91.40
Co-training (best)	92.20	94.20	93.20	93.20	92.60
Full SL	95.00	95.20	95.20	95.20	95.20

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9000 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

As shown in Table 17, CLM and BOW classifiers together produced the highest baseline and fully supervised learning performance for movie review data when compared to previous co-training runs. All measures of classification accuracy for co-training runs using these two classifiers exceeded 90%: With only 100 labeled sentences, this co-training strategy produced a classification accuracy of 92.20%, which is higher than all other results reported for fully supervised learning runs using either one or two classifiers. (The closest fully supervised learning run produced 92% classification accuracy using 9200 labeled data with a Naïve Bayes classifier (see Table 4)).

In order to understand whether these two different classifiers actually helped each other during iterations, individual performance for CLM and BOW classifiers was analyzed. When comparing the performance of the BOW classifier

during co-training iterations to the performance of corresponding self-training runs, the former using both CLM and BOW always outperformed the latter using BOW classifier alone, indicating that the BOW classifier was helped by the CLM classifier. Similarly, the CLM classifier also gained benefits from the BOW classifier during co-training.

Table 18

Top Features Generated by Converged CLM and BOW Classifiers

Features Type	CLM Classifier	BOW Classifier
Opinion Features	_h, ti, hi, ha, ou, _d, le, ve, her_, _it, _wh, n_t	but, it's, not, more, than, like, 'the film', so, just, even, most, no, 'the movie', good, characters, enough, 'this is', 'a movie', funny, director, 'it's a', isn't, made, would any ,'of its'
Non-opinion Features	#,), (, &, ;, ", er_, his_, n_the, _,_a, s_to_, he, r_, ing_the_, _from, j, ith_, ation_	'in the', life, story, 'on the', new, love, young, after, man, world, 'with the', 'of his', find, family, back, 'the story', finds, father, 'and his', friends, girl, 'story of', 'a young', woman

Note. Space in character n-grams is replaced by underscore.

As shown in Table 18, while the BOW classifier selected features that appear to be semantically or syntactically related to opinion or non-opinion classes (e.g., "funny", "in the"), the CLM classifier selected patterns that appear to be morphologically or symbolically related to opinion or non-opinion classes (e.g., "hi", "(") and sometimes syntactically relevant (e.g., "_d", "ing_the_").

Table 19

Classification Accuracy (%) of Co-Training for News Articles Where One Classifier Uses CLM with Character 8-grams and the Other Uses BOW Model with Word Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL	65.84	65.84	70.04	68.70	71.95
Co-training (converged)	68.51	68.51	70.99	73.66	72.52
Co-training (best)	70.03	69.66	71.57	74.43	74.24
Full SL	81.49	80.92	81.11	82.06	81.87

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 9270 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

Table 20

Classification Accuracy (%) of Co-Training for Blog Posts Where One Classifier Uses CLM with Character 8-grams and the Other Uses BOW Model with Word Unigrams and Bigrams

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	53.40	55.60	58.80	60.20	64.20
Co-training (converged)	53.60	53.60	57.40	58.20	59.80
Co-training (best)	56.60	57.00	60.60	62.40	66.00
Full SL	72.40	72.80	73.20	73.40	73.80

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL runs used an additional 7740 labeled sentences. SL runs involved two classifiers using a simple voting strategy.

As shown in Table 19 and Table 20, co-training with CLM and BOW classifiers improved on the supervised learning baseline only slightly when dealing with news articles but showed no improvement in the blog domain.

5.3.5 Other Co-Training Strategies

Motivated by the success of using non-words features, a co-training experiment was conducted to evaluate the use of part-of-speech (POS) tags as features for opinion detection. POS tags are word classes such as noun, verb, adjective, etc., and are the fundamental elements in linguistics analysis. Since POS tagging is computationally expensive and because POS tags were included in the Penn Treebank (Marcus, Marcinkiewicz & Santorini, 1993), only news articles were tested for BOW and POS co-training. The results suggested that POS tags alone were not strong features for sentence-level opinion detection and that more sophisticated linguistic features should be investigated in the future.

Co-training using two different feature selection methods (i.e., information gain and chi square) was also investigate; but the results indicated that these strategies were not productive for co-training because the top features selected by both methods were similar.

5.3.6 Summary of Co-Training Strategies

Different co-training strategies were evaluated over three data domains. Overall, co-training strategies were effective in classifying opinion sentences in the movie review domain, but they showed only limited value in the news domain and no benefits in the blog domain.

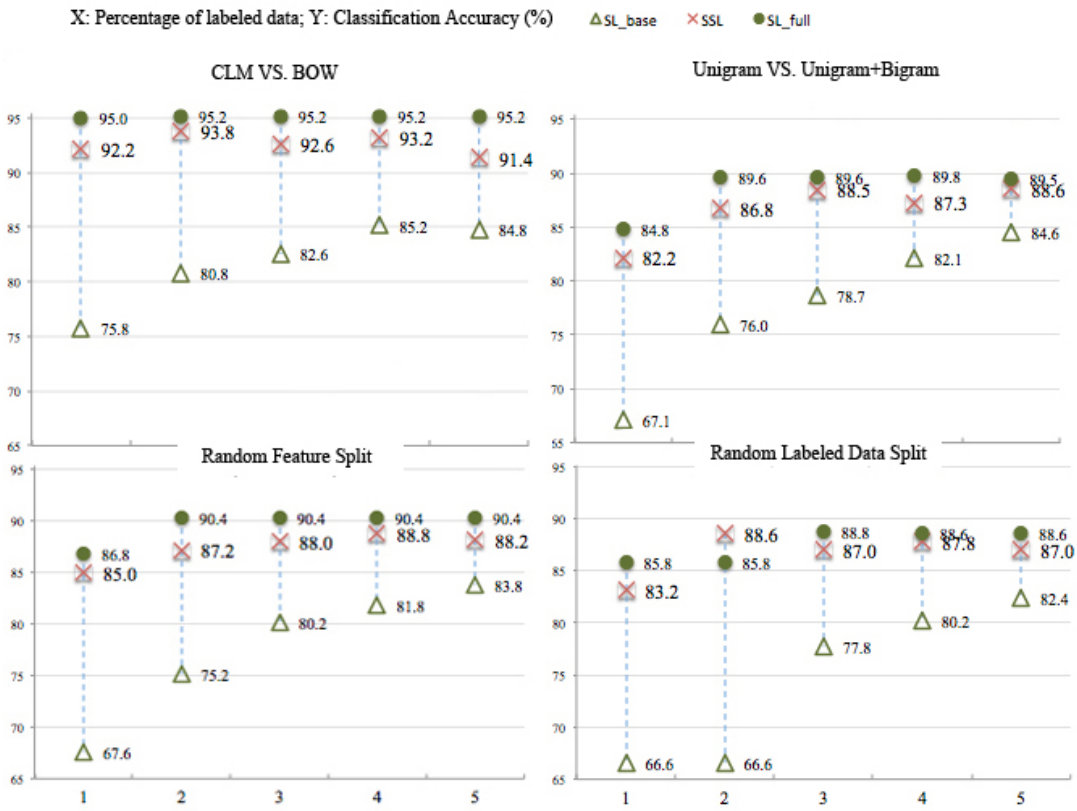

Figure 7. Performance of four co-training strategies for movie review data.

Figure 7 compares the performance of four co-training strategies for the movie review domain. For each chart in Figure 7, the x axis represents the percentage of labeled data (i.e., from 1% to 5%); the y axis represents classification accuracy; triangles indicate baseline SL runs; circles indicate full SL runs; and Xs indicate SSL runs. Numbers next to symbols reflect classification accuracy. For each line, an X located above a triangle indicates that the SSL run improved on the SL baseline: the closer the X is to the circle, the more effective was the SSL run.

Clearly, co-training using CLM and BOW classifiers achieved the best performance (see the chart in the upper left quadrant of Figure 7). Co-training

experiments based on different feature sets had very similar performance, as indicated by comparison of the charts in the upper right and lower left quadrants of Figure 7. Although co-training with random split labeled data had the lowest baseline performance (see the chart in the lower right quadrant), converged performance was as high as feature split co-training. This demonstrates the robustness and potential of co-training for opinion detection.

5.4 Compare Co-Training with Other SSL Methods

Comparison of co-training with self-training further illustrates the effectiveness of co-training in terms of overall performance and the number of labeled data needed for optimized performance, as demonstrated when the best co-training method is compared with self-training, EM-NB and S^3VM.

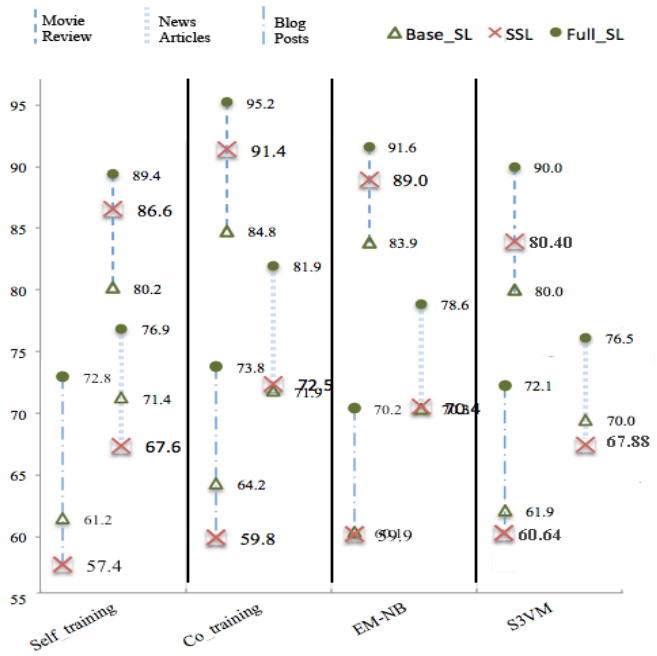

Figure 8. Classification accuracy (%) of SSL and SL in three datasets (i=5).

For all three datasets, Figure 8 demonstrates the performance of four types of SSL runs relative to the corresponding baselines and full SL runs. All SSL runs reported here used 5% data as labeled data. Lines with different patterns represent different datasets; triangles indicate baseline SL runs; circles indicate full SL runs; Xs indicate SSL runs; and numbers next to symbols reflect classification accuracy. From movie reviews to news articles to blog posts, the classification accuracy of baseline SL runs decreased as did the improvement in SSL runs. With greater than 80% baseline accuracy on movie reviews, SSL runs were most effective and showed trends similar to SSL for traditional topical classification (Nigam & Ghani, 2000); with slightly more than 70% baseline accuracy on news articles, self-training actually decreased performance of the corresponding SL baseline, while co-training and EM-NB outperformed the SL baseline only slightly; and, with approximately 60% baseline accuracy on blog posts, none of the SSL methods showed improvement.

Overall, for movie reviews and news articles, co-training proved to be most robust and effective, and EM-NB showed consistent but limited improvement over the SL baseline. An examination of EM-NB iterations for movie reviews shows that, with only 32 labeled sentences, EM-NB was able to achieve 88% classification accuracy, which is close to the best performance of simple Naïve Bayes self-training using 300 labeled sentences. For news articles, EM-NB increased accuracy from 63.5% to 68.8% with only 100 labeled sentences. This indicates that the problem space of opinion detection may be successfully described by the mixture model assumption of EM.

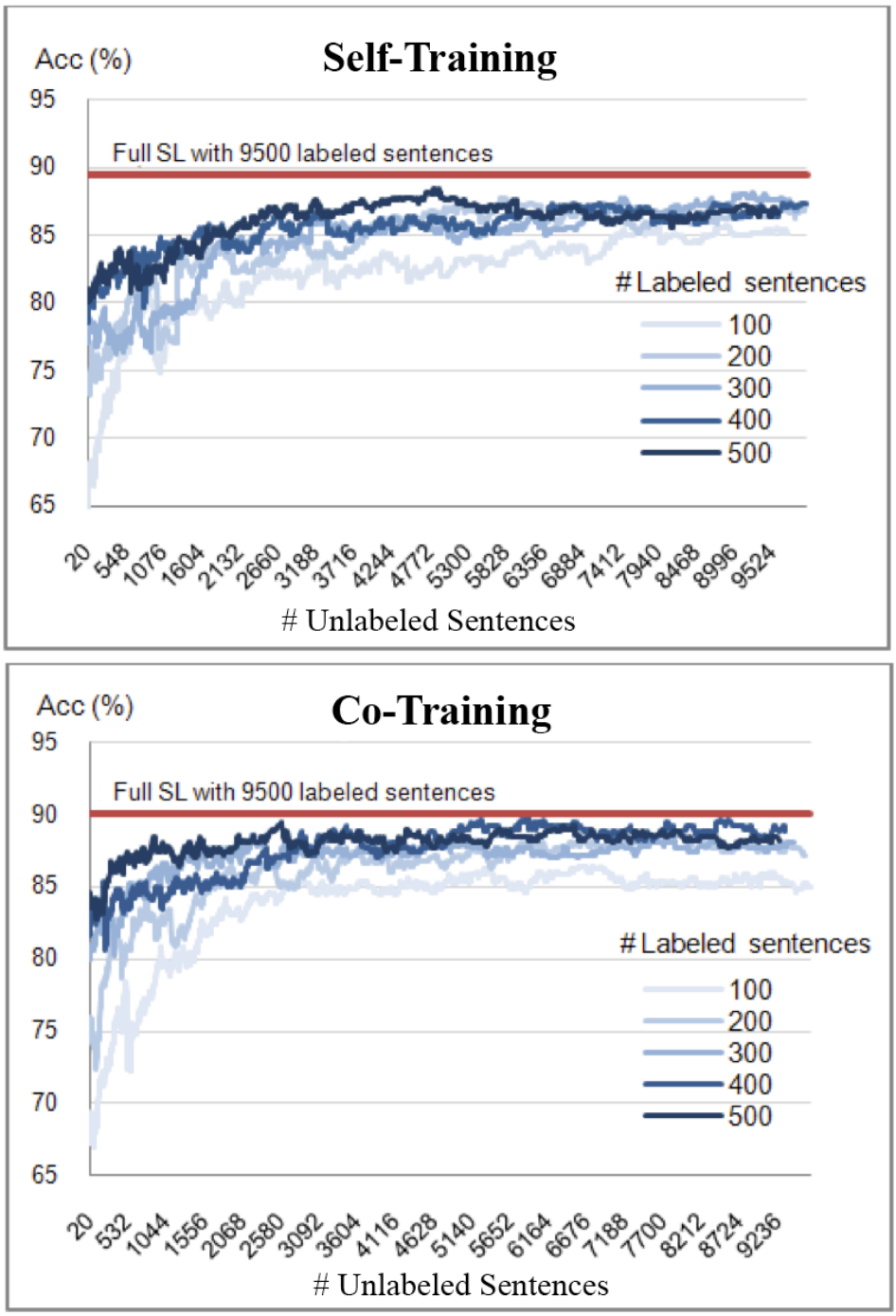

Figure 9. Performance of self-training and co-training over iterations.

Observation of the performance of self-training and co-training over iterations confirms that co-training uses labeled data more effectively for opinion

detection than self-training, as has been found in traditional topical classification (Nigam & Ghani, 2000). Figure 9 illustrates the performance of self-training and co-training over time. Straight lines correspond to full SL runs and are asymptotic for the curved lines representing SSL runs. Overall, co-training runs performed better than self-training runs as demonstrated by the fact that their lines are closer to the corresponding straight line. Co-training runs also reached optimized performance faster, given that their lines approach the full SL line more quickly. For example, with 500 labeled sentences, a self-training run reaches an optimized classification accuracy of 88.2% after labeling 4,828 sentences, while the co-training run reaches its optimized performance of 89.4% after labeling only 2,588 sentences.

5.5 Domain Adaptation

SSL methods did not perform well for the JDPA blog data and showed only minimal improvement over baseline SL in the news domain. One reason for the failure of SSL in these domains is the low classification accuracy of initial runs: The performance of blog baseline classifiers was only slightly better than chance (50%) and decreased the quality of auto-labeled data. In order to deal with challenging data domains such as blog posts, one possible solution is to improve baseline accuracy for SSL by introducing high-quality features: for example, augmenting the feature set with domain independent opinion lexicons such as those which have been suggested as effective in creating high precision opinion classifiers. An alternative approach for dealing with challenging data domains is

to borrow labeled data from one ore more "easy" domains: for example, the use of movie review data in SSL applications for opinion detection in news article and blog domains. Self-training was adopted to investigate both of these approaches to handling challenging data through domain adaptation because it is a fundamental SSL method that is easy to generalize.

5.5.1 Using Domain-Independent Opinion Lexicons

In addition to unigram and bigram features with binary values, nine lexicon features were added to the feature set. To avoid the possibility that the large number of n-gram features would weaken these nine lexicon features, the value of each lexicon feature (e.g., dynamic adjective) was not binary but represented the total number of matches between lexicon terms and the words in a target sentence. For example, the value of Wilson lexicon features for the sentence "I like these two much better than the versions made for Hong Kong market" is two because two Wilson lexicon terms, 'like' and 'better', are used in this sentence. Redundancy between lexicons was not removed under the assumption that one word occurring in multiple lexicons makes it a strong opinion indicator. For example, 'like' is included in the Levin verb class lexicon, the frameNet lexicon and the Wilson lexicons, and its occurrences were counted when calculating values for all three lexicon features.

Table 21

Classification Accuracy (%) of Self-Training With and Without Opinion Lexicon Features for News Articles

	# of Original Labeled Sentences				
Run Type	103	206	309	412	515
Baseline SL w/o Lexicon	60.50	64.31	69.47	69.47	71.38
Self-training w/o Lexicon	60.11	65.84	66.41	67.75	67.56
Baseline SL w/ Lexicon	66.60	70.42	70.99	72.14	72.52
Self-training w/ Lexicon	59.73	66.41	71.18	70.61	70.61

Note. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams.

Table 22

Classification Accuracy (%) of Self-Training With and Without Opinion Lexicon Features for Blog Posts

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL w/o Lexicon	55.05	58.95	61.93	64.69	66.06
Self-training w/o Lexicon	54.59	55.73	56.65	58.49	64.45
Baseline SL w/ Lexicon	63.76	64.68	63.53	66.51	67.89
Self-training w/ Lexicon	51.38	62.16	55.73	61.47	69.04

Note. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams.

In Table 21 and Table 22, the baseline supervised learning runs using domain-independent opinion lexicon features (i.e., Baseline SL w/ Lexicon) produced higher classification accuracies than baseline supervised learning runs that did not use lexicon features (i.e., Baseline SL w/o Lexicon). However, self-training runs that used opinion lexicons (i.e., Self-training w/ Lexicon) did not generally improve the baseline run (i.e., Baseline SL w/ Lexicon); in some cases,

performance was even lower than that of the corresponding self-training runs that did not use domain-independent opinion lexicon information (i.e., Self-training w/o Lexicon). For example, using opinion lexicon features with 86 labeled blog sentences, supervised learning yielded a classification accuracy of 63.76%, 8.71% higher in absolute value than the classification accuracy produced by the supervised learning run that made no use of opinion lexicon features; however, after self-training iterations, the performance of the former run decreased to 51.38%, 3.21% lower in the absolute value of classification accuracy than the classification accuracy produced by the latter run. This may be because, as a closer look at the distribution of opinion lexicon terms in the three datasets indicates, many opinion lexicon terms actually occur frequently in objective, non-opinion, sentences.

Table 23

Distribution of Domain Independent Opinion Lexicons

	Dataset					
	Movie Reviews		News Articles		Blog Posts	
# of matches	Non-Op	Op	Non-Op	Op	Non-Op	Op
Unique Terms	1076	1428	502	1127	459	753
Total Occurrence	4867	8596	1865	5576	1778	4668

Note. Non-Op=non-opinion; Op=opinion.

Table 23 shows the number of unique opinion lexicon terms that appear in subjective and objective data in the three data domains as well as the total occurrence of opinion lexicon terms in subjective and objective sentences. Although opinion lexicon terms are used more often in opinion sentences than in

non-opinion sentences, their presence does not appear to be a strong indicator of opinions. For example, more than half of the opinion lexicon features that appear in opinion blog sentences also appear in non-opinion blog sentences. When considering their total occurrence, opinion lexicon terms are used in opinion sentences approximately three times as often as in non-opinion sentences in both the blog and news data domains; opinion lexicon terms are used in non-opinion sentences a little more than half as often as in opinion sentences in the movie review domain. This suggests that automatically created subjective and objective movie review data will not necessarily reflect opinion and non-opinion classes.

The inefficiency of opinion lexicons can be attributed to the fact that opinion features are often very sensitive to the context in which they occur. For example, “like” is included in three opinion lexicons and is therefore treated as a good opinion indicator, but, when it is used in the sentence “the lens cap finally snaps into the front of the lens like other maker's models”, it is no longer an opinion indicator.

When looking at individual opinion lexicons (see Table 24), the blog domain is the most difficult domain for opinion lexicon application. Even the IU collocation, which has been demonstrated by Yang et al. (2007) as the most effective features for the opinion detection task in the context of the Blog track, appears to have very weak power for differentiating subjective and objective sentences.

Table 24

Distribution of Domain Independent Opinion Lexicons:
by Source of Lexicon

	IU	Levin	N-gram	Appr.	Colin	Dyn.	Fnet	IGI	Wilson
M_nop	2	125	59	521	1	1	111	743	890
M_op	41	163	87	712	1	1	162	911	1131
N_nop	21	67	57	210	0	0	59	308	324
N_op	45	155	75	469	0	0	149	745	861
B_nop	73	63	65	171	0	0	57	193	245
B_op	73	81	97	344	0	0	89	391	467

Note. M=Movie reviews; N=News articles; B=Blog posts; nop=non-opinion; op=opinion. IU=IU collocation; Levin=Levin's verb class; N-gram=review bigrams; Appr=Appraisal; Colin=Colin's adjectives; Dyn=Dynamic adjectives; Fnet=FrameNet.

One explanation for the less than satisfactory contribution of domain-independent opinion lexicons is that, when there was a limited number of labeled data at the beginning of an SSL run, extra opinion lexicon features helped; however, with more and more unlabeled data labeled automatically and used to replenish the labeled dataset, the limitations of opinion lexicons were amplified, undermining overall performance.

5.5.2 Using Labeled Data in Non-Target Domain

A preliminary experiment on the use of movie review data was conducted on the news domain. This analysis was followed by a more in-depth investigation of the use of movie review data in the blog data domain.

5.5.2.1 From Movie Reviews to News Articles

This experiment tested an extreme situation where there was no labeled data available in the target data domain. To begin, 9,500 labeled movie review sentences were used to train a Naïve Bayes classifier. Although this classifier produced a fairly good classification accuracy of 89.2% on movie review data, its accuracy in a domain-transfer SL run on news data was poor (64.1%), demonstrating the severity of the domain transfer problem.

A self-training run starting with the same Naïve Bayes classifier trained on movie review data and using unlabeled data from the news domain (i.e., a domain-transfer SSL run) showed some improvement, achieving a classification accuracy of 75.1% that surpassed the domain-transfer SL run by more than 17%. To further understand how well SSL handles the domain transfer problem, a full SL run that used all labeled news sentences was also performed. This full SL run achieved 76.9% classification accuracy, only 1.8% higher in absolute value than the domain-transfer SSL run, which had not used any labeled news data.

5.5.2.2 From Movie Reviews/News to Blog Posts

The preliminary domain transfer experiment using movie reviews with the news articles dataset indicated the potential of SSL methods for domain adaptation. This approach was applied to the blog dataset to investigate the value of self-training and co-training when dealing with domain adaptation in challenging domains.

Domain Transfer Self-training

Domain transfer self-training runs for blog data combined all movie review data and i% labeled blog data to form the initial labeled dataset and then followed the traditional self-training procedure. A control factor was introduced and investigated to gradually reduce the impact of out-of-domain data (i.e., movie reviews) on each iteration.

Table 25

Classification Accuracy (%) of Self-training With and Without Labeled Movie Reviews

	# of Original Labeled Sentences (Blog)				
Run Type	86	172	258	344	430
Baseline SL w/o Mreview	55.05	58.95	61.93	64.69	66.06
Self-training w/o Mreview	54.59	55.73	56.65	58.49	64.45
Baseline SL w/ Mreview	63.07	62.16	62.61	62.16	61.70
Self-training w/Mreview	71.10	70.87	71.41	70.41	71.10
Self-training w/Mreview w/weight control	72.94	72.94	72.48	71.56	71.79
Full SL	71.56	73.17	72.71	72.94	72.48

Note. Mreview = Movie reviews. Settings for self-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. Full SL used an additional 7740 labeled blog sentences.

Table 25 reports the results of self-training runs to identify opinion sentences in blog posts, both with and without the use of movie review data, as well as corresponding baseline and fully supervised learning runs. The results for baseline SL runs without movie reviews and self-training without movie reviews show that

self-training using only blog data decreases baseline SL performance. By keeping the same settings and adding more labeled data from the movie review domain, self-training with movie reviews increased the performance of SL runs by 12% to 15% and came closer to the performance of full SL runs, which used 90% of the labeled blog data. In the case of domain transfer runs, the number of available in-domain labeled data did not appear to have an impact on overall performance: Neither supervised nor semi-supervised runs using movie review data produced higher classification accuracies with increasing numbers of labeled blog sentences. For example, the self-training run using movie review data yielded a classification accuracy of 71.10% with as few as 86 or as many as 430 labeled blog sentences in the original training set. This may be due to the preponderance of movie review data available during training.

A control factor intended to reduce the bias of movie review data was added to weaken the effects of domain transfer gradually (i.e., a decrease of 0.001 on each iteration). The results reported for self-training runs with both movie review data and weight control show that these runs outperformed runs that did not use weight control by 1% to 3%, reaching and occasionally exceeding the performance of the full SL run.

Domain Transfer Co-training

All co-training strategies investigated in this research could be adapted to deal with the domain transfer problem by adding labeled movie review sentences to the initial labeled dataset. In a final experiment, one co-training strategy was designed and tested specifically for domain adaptation, assuming that two

classifiers using labeled data from different non-target data domains could counteract the bias of each domain. The results of this strategy are reported in Table 26 and show no benefit from applying this domain transfer co-training strategy. For example, with 430 labeled blog sentences, co-training with one classifier using 90% of total movie review data and the other classifier using 90% of total news article data achieved classification accuracy of 59.63%, 7.8% lower than the baseline supervised learning run, which had not used any unlabeled blog data. Since movie review data alone has proven effective for classifying blog data in domain transfer self-training, the classifier that used news articles may have been responsible for the failure of domain transfer co-training.

Table 26

Classification Accuracy (%) of Co-Training Where One Classifier is Trained on Movie Reviews and the Other on News Articles.

	# of Original Labeled Sentences				
Run Type	86	172	258	344	430
Baseline SL	58.26	65.14	64.68	67.20	67.43
Co-training	59.63	61.93	63.53	54.40	59.63
Full SL	73.17	72.71	73.62	73.62	73.85

Note. Settings for co-training: u=20, p=2, n=2, n-grams=unigrams+bigrams. SL runs involved two classifiers with a simple voting strategy. Full SL used an additional 7740 labeled blog sentences for each classifier.

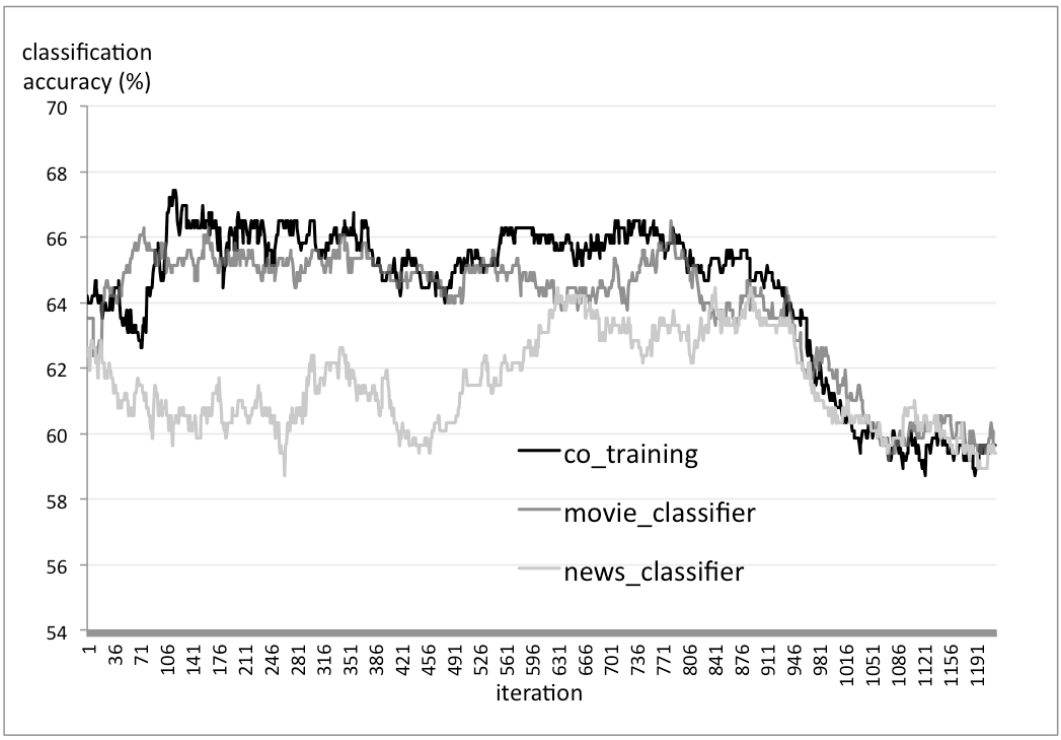

*Figure 10***.** Classification accuracy (%) for domain transfer co-training (i=5).

The visualization in Figure 10 appears to confirm this analysis. The lower line represents the performance over iterations of the classifier initially trained by news article data, the middle line indicates the performance of the classifier initially trained by movie review data, and the upper line marks the performance of the composite of those two classifiers (i.e., the performance of co-training). The performance of the news classifier appears to have decreased initially because the news domain is challenging for opinion detection, and errors made in early iterations were reinforced during later iterations, eventually pulling down overall performance.

5.5.3 Summary of Domain Adaptation Experiments

For challenging data domains, adoption of domain independent opinion lexicons resulted in only minimal improvement, but applying simple self-training alone was promising for tackling domain transfer from the source domain of movie reviews to the target domains of news articles and blog posts. However, a co-training strategy that aimed to overcome the domain transfer problem by using two non-target domains was not successful because of the poor performance of a classifier trained on one of the non-target news article domain.

5.6 Results Summary

Overall, the results of this series of experiments suggest that SSL improves accuracy for opinion detection although its contribution varies across data domains, and different strategies need to be applied, based on the data domain, to achieve optimized performance. For the movie review dataset, SSL runs generally outperformed their corresponding baseline SL runs and approached the level of classification accuracy of full SL runs. For the news article dataset, SSL performance followed a similar trend but demonstrated only a small rate of increase. For the blog post dataset, SSL runs using only blog data showed no improvement over the SL baseline; however, with the use of labeled movie review data, SSL runs produced results comparable with the results of full SL.

6 Conclusion

The growth of Web 2.0 applications makes it easy for Internet users to share their knowledge, experience, and opinions, and this easily accessible and widely available user-generated content provides a rich resource for decision making.

Opinion mining is a research area that has developed at the intersection of text mining and natural language processing. It consists of subareas that include opinion identification, opinion summarization, and the understanding of different characteristics of opinion expressions. Because opinion expressions in user-generated content are usually mixed with non-opinion expressions, the fundamental task for opinion mining is opinion identification.

Prior research has suggested that a large number of opinion-bearing features are necessary for capturing subtle opinions because opinions can be expressed using a wide range of words or patterns. However, researches often face the challenge of limited amounts of opinion-labeled data from which opinion-bearing features can be extracted, especially at the sentence level. This shortage of labeled data has become a severe challenge for developing effective opinion detection systems. Since opinion is an important aspect of many types of information and being able to identify and organize opinion is essential for information studies, the research reported here tackled this challenge by investigating semi-supervised learning (SSL) algorithms, motivated by limited labeled data and the availability of plentiful unlabeled data.

Four SSL algorithms based on different assumptions were examined: self-training, co-training, Expectation Maximization with Naïve Bayes (EM-NB) and Semi-Supervised Support Vector Machines (S^3VM). These algorithms were applied to three datasets from domains with different characteristics (i.e., movie reviews, news articles and blog posts), and their performance varied across domains. For movie reviews, all SSL methods except S^3VM showed the advantage of using unlabeled data for opinion detection, and co-training strategies attained state-of-the-art results with a small number of labeled sentences. Due to the nature of the movie review data, opinion detection in movie reviews is an "easy" problem because it involves genre classification and thus relies, strictly speaking, on distinguishing movie reviews from plot summaries.

For other manually created datasets that are expected to reflect real opinion characteristics, the SSL approach was impeded by low baseline precision and demonstrated only limited improvement. For news articles, EM-NB and co-training were able to achieve slight improvement in performance over supervised learning using only labeled data. Blog posts are the most challenging domain and blog data, showed no benefits from implementing any SSL methods. However, with the addition of out-of-domain labeled data (i.e., movie review data), self-training for identifying opinion sentences in blogs exceeded fully supervised learning using all available labeled blog data.

Opinion detection is a growing research area with many potential applications (e.g., personalized search, metadata labeling, business intelligence). Thus the contributions of this research are four-fold. First, the findings of this

research indicate a general approach that can be adapted for use in existing opinion detection or sentiment analysis systems across data domains and across languages[28]. These findings also provide valuable guidelines and evaluation baselines for later studies applying SSL algorithms in opinion detection. Second, there are several applications for automatically labeled data generated by the effective SSL strategies reported in this research: creating opinion annotated corpora directly; providing candidates for manual annotation; and extracting opinion-bearing features. Third, the SSL strategies investigated in this research, especially those related to domain adaptation, are readily extensible to other text mining systems (e.g., genre identification). Finally, this research contributes to SSL research by expanding the spectrum of SSL applications to include opinion mining, confirming the effectiveness of SSL as a general approach for dealing with insufficient labeled data, and providing successful new co-training strategies and approaches for domain adaptation.

Although spam detection was not addressed in the current research, it will become a crucial aspect in future opinion mining research: as more and more people seek online opinions, more and more spam advertisements will be published on the Web (e.g., spam blog posts, spam review comments). Because spam detection systems generally require labeled data, they will face problems of domain transfer, and the SSL strategies investigated in this research can be easily adapted for this future direction.

[28] SSL has successfully solved NLP tasks in non-English domains such as Chinese (Wang, Huang, & Harper, 2007).

In the future of opinion mining research, the greatest successes are likely to come from strategies that can integrate different sources of evidence and leverage both human knowledge and machine techniques. Traditional statistical methods and simple bag-of-words features will continue to play important roles in dealing with the ever-growing body of content on the Web; natural language processing techniques and a large range of linguistic features will be advantageous in capturing subtle opinion cues, especially in structured scholarly data; and everyday knowledge, or common sense, will assist in the identification of implicit opinions, including sarcasm and irony. While the shortage of labeled data will continue to be a challenge for opinion mining and may become even more severe in emerging data domains, the SSL methods investigated in this research, especially co-training and the domain-transfer strategies, are important for hybrid opinion mining.

7 References

Alias-i. (2008). LingPipe 4.0.1. http://alias-i.com/lingpipe (accessed October 1, 2008)

Abbasi, A., Chen, H., & Salem, A. (2008). Sentiment analysis in multiple languages: Feature selection for opinion classification in Web forums. *ACM Transactions on Information Systems, 26*(3).

Abney, S. P. (2002). Bootstrapping. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, (pp. 360-367).

Abney, S. P. (2008). Semisupervised learning for computational linguistics. *Computational Linguistics*, 34(3), 449-452.

Agrawal, R., Rajagopalan, S., Srikant, R., & Xu, Y. (2003). Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th International Conference on World Wide Web*, May 20-24, 2003, Budapest, Hungary (pp. 529 535). New York, NY: ACM.

Aue, A., & Gamon, M. (2005). Customizing sentiment classifiers to new domains: A case study. In *Proceedings of International Conference Recent Advances in Natural Language Processing (RANLP-2005)*, September 21-23, 2005, Borovets, Bulgaria.

Bethard, S., Yu, H., Thornton, A., Hatzivassiloglou, V., & Jurafsky, D. (2006). Extracting opinion propositions and opinion holders using syntactic and lexical cues. In J. G. Shanahan, Y. Qu & J. Wiebe (Eds.), *Computing attitude and affect in text: Theory and applications* (Vol. 20, pp. 125-141).

Bengio, Y., Delalleau, O., & Roux, N. L. (2006). The curse of highly variable functions for local kernel machines. *Advances in Neural Information Processing Systems* 18.

Bennett, K. P., & Demiriz, A. (1998). Semi-supervised support vector machines. In *Proceedings of the 1998 conference on Advances in neural information processing systems*, (pp. 368-374).

Blitzer, J., Dredze, M., & Pereira, F. (2007). Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, (pp. 440-447). Association for Computational Linguistics.

Bloom, K., Garg, N., & Argamon, S. (2007). Extracting appraisal expressions. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT)*, Rochester, NY (pp. 308-315). Morristown, NJ: Association for Computational Linguistics.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, (pp. 92-100).

Breck, E., Choi, Y., & Cardie, C. (2007). Identifying expressions of opinion in context. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, January 6-12, Hyderabad, India (pp. 2683-2688).

Bruce, R., & Wiebe, J. (1999). Recognizing subjectivity: A case study in manual tagging. *Natural Language Engineering, 5*, 187-205.

Cao, Y., Li, H., & Lian, L. (2003). Uncertainty reduction in collaborative bootstrapping: measure and algorithm. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Sapporo, Japan (pp. 327-334).

Chapelle, O., Scholkopf, B., & Zien, A. (2006). *Semi-supervised Learning*: The MIT Press.

Chapelle, O., Weston, J., & Scholkopf, B. (2003). Cluster kernels for semi-supervised learning. In S. T. S. Becker, and K. Obermayer (Ed.), *Advances in Neural Information Processing Systems* (Vol. 15, pp. 585-592): MIT Press

Chesley, P., Vincent, B., Xu, L., & Srihari, R. K. (2006). Using verbs and adjectives to automatically classify blog sentiment. In *Proceedings of AAAI-CAAW-06, the Spring Symposia on Computational Approaches to Analyzing Weblogs,* March 27-29, 2006, Stanford University, CA. Menlo Park, CA: AAAI Press.

Chklovski, T. (2006). Deriving quantitative overviews of free text assessments on the Web. In *Proceedings of the 11th International Conference on Intelligent User Interfaces*, Sydney, Australia (pp. 155-162). New York, NY: ACM.

Clark, S., Curran, J. R., & Osborne, M. (2003). Bootstrapping POS taggers using unlabelled data. In *Proceedings of the Seventh Conference on Computational Natural Language Learning (CoNLL-03)*, Edmonton, Canada.

Collins, M., & Singer, Y. (1999). Unsupervised models for named entity classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, (pp. 100-110).

Conrad, J. G., & Schilder, F. (2007). Opinion mining in legal blogs. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, Stanford, CA (pp. 231-236). New York, NY: ACM.

Constant, N., Davis, C., Potts, C., & Schwarz, F. (2009). The pragmatics of expressive content: Evidence from large corpora. *Sprache und Datenverarbeitung* (Vol. 33, pp.5-21).

Cui, H., Mittal, V. O., & Datar, M. (2006). Comparative experiments on sentiment classification for online product reviews. In *Proceedings of the 21st National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, July 16-20, 2006, Boston, MA (pp. 1265-1270). Menlo Park, CA: AAAI Press.

Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of WWW'03: The 12th International Conference on World Wide Web*, May 20-24, 2003, Budapest, Hungary (pp. 519-528). New York, NY: ACM Press.

Dasgupta, S., & Ng, V. (2009). Topic-wise, sentiment-wise, or otherwise? Identifying the hidden dimension for unsupervised text classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, Singapore (pp. 580-589).

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39, 1-38.

Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the International Conference on Web Search and Web Data Mining*, Feb 11-12, 2008, Palo Alto, CA (pp. 231-240). New York, NY: ACM.

Eckle-Kohler, J., Kohler, M., & Mehnert, J. (2005). Automatic recognition of German news focusing on future-directed beliefs and intentions. *Computer Speech and Language, 22*(4), 394-414.

Efron, M. (2004). Cultural orientation: Classifying subjective documents by cocitation analysis. In *Proceedings of AAAI Fall Symposium on Style and Meaning in Language, Art, and Music*, October 21-24, Washington, D.C. (pp. 41-48).

Esuli, A., & Sebastiani, F. (2006). Determining term subjectivity and term orientation for opinion mining. In *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics (EACL 2006)* April 3-7, 2006, Trento, Italy (pp. 193-200).

Feng, H., & Chua, T.-S. (2003). A bootstrapping approach to annotating large image collection. In *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, Berkeley, California (pp. 55-62).

Fillmore, C. J., & Baker, C. F. (2001). Frame semantics for text understanding. In *Proceedings of WordNet and Other Lexical Resources Workshop*, Pittsburgh, PA.

Gamon, M. (2004). Sentiment classification on customer feedback data: Noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the 20th International Conference on Computational*

Linguistics, August 23-27, 2004, Geneva, Switzerland. Morristown, NJ: Association for Computational Linguistics.

Gamon, M., & Aue, A. (2005). Automatic identification of sentiment vocabulary: Exploiting low association with known sentiment terms. In *Proceedings of the ACL Workshop on Feature Engineering for Machine Learning in NLP*, Ann Arbor, MI (pp. 57-64).

Gamon, M., Aue, A., Corston-Oliver, S., & Ringger, E. (2005). Pulse: Mining customer opinions from free text. In *Advances in Intelligent Data Analysis VI* (pp. 121-132).

Glance, N., Hurst, M., Nigam, K., Siegler, M., Stockton, R., & Tomokiyo, T. (2005). Deriving marketing intelligence from online discussion. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, August 21-24, 2005, Chicago, IL (pp. 419-428). New York, NY: ACM.

Goldman, S., & Zhou, Y. (2000). Enhancing supervised learning with unlabeled data. In *Proceedings of the 17th International Conference on Machine Learning*, San Francisco, CA (pp. 327-334).

Goutte, C., Déjean, H., Gaussier, E., Cancedda, N., & Renders, J.-M. (2002). Combining labelled and unlabelled data: A case study on fisher kernels and transductive inference for biological entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning*, (pp. 1-7). Association for Computational Linguistics.

Grefenstette, G., Qu, Y., Shanahan, J. G., & Evans, D. A. (2004). Coupling niche browsers and affect analysis for an opinion mining application. In *Proceedings of the 7th International RIAO Conference (Recherche d'Information Assistée par Ordinateur)*, April 26-28, Avignon, FR (pp. 186-194).

Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics*, July 07-12, 1997, Madrid, Spain (pp. 174-181).

Hatzivassiloglou, V., & Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th Conference on Computational Linguistics*, July 31-August 04, 2000, Saarbrücken, Germany (pp. 299-305). Morristown, NJ: Association for Computational Linguistics.

Hancock, J. T., Landrigan, C., & Silver, C. (2007). Expressing emotion in text-based communication. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, San Jose, CA (pp. 929-932). New York, NY: ACM.

Holzman, L. E., & Pottenger, W. M. (2003). *Classification of emotions in Internet chat: An application of machine learning using speech phonemes*. (P. L. U. Bethlehem No. Document Number)

Hsu, C., Chang, C., & Lin, C. (2003). *A practical guide to support vector classification* (Technical Report). National Taiwan University, Department of Computer Science and Information Engineering.

Hurst, M., & Nigam, K. (2004). Retrieving topical sentiments from online document collections. In *Proceedings of the 11th Conference on Document Recognition and Retrieval*, January 21-22, 2004, San Jose, CA (pp. 27-34).

Jin, W., Ho, H. H., & Srihari, R. K. (2009). OpinionMiner: A novel machine learning system for Web opinion mining. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France (pp. 1195-1204).

Joachims, T. (1999a). Making large-Scale SVM Learning Practical. In B. Schölkopf, C. J. C. Burges & A. J. Smola (Eds.), *Advances in Kernel Methods Support Vector Learning*: MIT Press.

Joachims, T. (1999b). Transductive inference for text classification using support vector machines. In Proceedings of the International Conference on Machine Learning.

Joachims, T. (2001). A statistical learning model of text classification for support vector machines. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, LA (pp. 128-136). ACM Press.

Jurafsky, D., & Martin, J. H. (2008). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition* (2nd Ed.): Upper Saddle River, N.J.: Pearson Prentice-Hall.

Kale, A., Karandikar, A., Kolari, P., Java, A., Finin, T., & Joshi, A. (2007). Modeling trust and influence in the blogosphere using link polarity. In *Proceedings of the International Conference on Weblogs and Social Media Boulder*, Colorado, USA.

Kamps, J., Marx, M., Mokken, R. J., & Rijke, M. D. (2004). Using WordNet to measure semantic orientation of adjectives. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal (pp. 1115-1118).

Kanayama, H., & Nasukawa, T. (2006). Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, Sydney, Australia (pp. 355-363).

Karlgren, J., & Cutting, D. (1994). Recognizing text genres with simple metrics using discriminant analysis. In *Proceedings of the 15th Conference on*

Computational Linguistics, Kyoto, Japan (pp. 1071-1075). Morristown, NJ: Association for Computational Linguistics.

Kessler, J. S., Eckert, M., Clark, L., & Nicolov, N. (2010). The ICWSM 2010 JDPA sentiment corpus for the automotive domain. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media Data Workshop Challenge (ICWSM-DWC)*, Washington, D.C., USA.

Kim, S. M., & Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*, August 23-27, 2004, Geneva, Switzerland (pp. 1367-1373). Morristown, NJ: Association for Computational Linguistics.

Kim, S. M., & Hovy, E. (2005). Automatic detection of opinion bearing words and sentences. In *Companion Volume to the Proceedings of IJCNLP-05, the Second International Joint Conference on Natural Language Processing*, Jeju Island, Republic of Korea (pp. 61-66).

Kim, S. M., & Hovy, E. (2006). Extracting opinions, opinion holders, and topics expressed in online news media text. In *Proceedings of ACL/COLING Workshop on Sentiment and Subjectivity in Text*, July 22, 2006, Sydney, Australia (pp. 1-8). Morristown, NJ: Association for Computational Linguistics.

Koppel, M., & Shtrimberg, I. (2006). Good news or bad news? Let the market decide. In Y. Q. James G. Shanahan, Janyce Wiebe (Ed.), *Computing attitude and affect in text: Theory and applications* (pp. 297-301): Dordrecht: Springer.

Ku, L. W., & Chen, H.-H. (2007). Mining opinions from the Web: Beyond relevance retrieval. *Journal of the American Society for Information Science and Technology, 58*(12), 1838-1850.

Levin, B. (1993). *English verb classes and alternations*. Chicago, IL: University of Chicago Press.

Li, Y., Bontcheva, K., & Cunningham, H. (2007). Experiments of opinion analysis on two corpora MPQA and NTCIR-6. In *Proceedings of the 6th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-Lingual Information Access*, May 15-18, 2007, Tokyo, Japan (pp. 323-329).

Lin, W. H., Wilson, T., Wiebe, J., & Hauptmann, A. (2006). Which side are you on? Identifying perspectives at the document and sentence levels. In *Proceedings of Tenth Conference on Natural Language Learning (CoNLL'06)*, New York, US. Morristown, NJ: Association for Computational Linguistics.

Liu, B. (2007). *Web data mining: Exploring hyperlinks, contents and usage data (data-centric systems and applications)*. Berlin: Springer.

Liu, H., Lieberman, H., & Selker, T. (2003). A model of textual affect sensing using real-world knowledge. In *Proceedings of the 8th International Conference on Intelligent User Interfaces*, January 12-15, 2003, Miami, FL (pp. 125-132). New York, NY: ACM.

Maeireizo, B., Litman, D., & Hwa, R. (2004). Co-training for predicting emotions with spoken dialogue data. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, (pp. 203-206).

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

Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a large annotated corpus of English: The penn treebank. *Computational Linguistics, 19*(2), 313-330.

Marcus, M. P., Santorini, B., Marcinkiewicz, M. A., & Taylor, A. (1999). *Treebank-3* Linguistic Data Consortium, Philadelphia.

McDonald, R., Hannan, K., Neylon, T., Wells, M., & Reynar, J. (2007). Structured models for fine-to-coarse sentiment analysis. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, Prague, Czech Republic (pp. 432-439).

Macdonald, C., Ounis, I., & Soboroff, I. (2007). Overview of the TREC-2007 Blog track. *Proceedings of the 16th Text REtrieval Conference (TREC 2007)*.

Mishne, G. (2005). Experiments with mood classification in blog posts. In *Proceedings of Style2005 the 1st Workshop on Stylistic Analysis Of Text For Information Access*, at SIGIR 2005, Salvador, Bahia, Brazil.

Mullen, T., & Collier, N. (2004). Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain (pp. 412-418).

Nasukawa, T., & Yi, J. (2003). Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd International Conference on Knowledge Capture*, October 23-25, 2003, Sanibel Island, FL (pp. 70-77). New York, NY: ACM.

Ng, V., & Cardie C. (2003) Bootstrapping coreference classifiers with multiple machine learning algorithms. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)* (pp. 113-120).

Ng, V., Dasgupta, S., & Arifin, S. M. N. (2006). Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*, Sydney, Australia (pp. 611-618). Morristown, NJ: Association for Computational Linguistics.

Nigam, K., & Ghani, R. (2000). Analyzing the effectiveness and applicability of co-training. In *Proceedings of the Ninth International Conference on Information and Knowledge Management* (pp. 86-93).

Nigam, K., & Hurst, M. (2004). Towards a robust metric of opinion. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text*, March 22-24, 2004, Stanford, CA (pp. 598-603).

Nigam, K., & Hurst, M. (2006). Towards a robust metric of polarity. In *Computing attitude and affect in text: Theory and applications* (Vol. 20, pp. 265-279): Dordrecht, Netherlands: Springer.

Nigam, K., McCallum, A. K., Thrun, S., & Mitchell, T. (1999). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39, 103-134.

Niu, Y., Zhu, X., Li, J., & Hirst, G. (2005). Analysis of polarity information in medical text. In *Proceedings of the American Medical Informatics Association 2005 Annual Symposium*, October 22-26, 2005, Washington, DC (pp. 570-574).

Niu, Y. Z., Ji, D. H., & Tan, C.L. (2005). Word sense disambiguation using label propagation based semi-supervised learning. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics,* Jun 2005, Ann Arbor (pp. 395-402).

Opinion. (n.d.). In *Merriam-Webster's online dictionary*. Retrieved May 7, 2008, from http://www.merriam-webster.com/dictionary/opinion

Ounis, I., Macdonald, C., & Soboroff, I. (2008). Overview of the TREC-2008 Blog Track. In Proceeding of the17th Text REtrieval Conference (TREC 2008).

Ounis, I., Rijke, M. d., Macdonald, C., Mishne, G., & Soboroff, I. (2007). Overview of the TREC-2006 Blog track. *Proceedings of the 15th Text REtrieval Conference*.

Pang, B., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, July 21-26, 2004, Barcelona, Spain (pp. 271-278). Morristown, NJ: Association for Computational Linguistics.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval, 2*(1-2), 1-135.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, July 6-7, 2002, Philadelphia, PA (pp. 79-86). Morristown, NJ: Association for Computational Linguistics.

Pfeifer, R. (1988). Artificial Intelligence Models of Emotion. In V. Hamilton et al. (Ed.), *Cognitive Perspectives on Emotion and Motivation*. Kluwer Academic Publishers.

Pierce, D., & Cardie, C. (2001). Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, Pittsburgh, PA (pp. 1-9).

Picard, R. W. (1997). *Affective computing*. Cambridge, MA: The MIT Press.

Potts, C. & Schwarz, F. (2008). *Exclamatives and heightened emotion: Extracting pragmatic generalizations from large corpora*. manuscript, UMass Amherst.

Quirk, R., Greenbaum, S., Leech, G., & Svartvik, J. (1985). *A comprehensive grammar of the English language*. London: Longman.

Riloff, E., & Jones, R. (1999). Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, Orlando, FL (pp. 474-479).

Riloff, E., & Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, July 11-12, 2003, Sapporo, Japan (pp. 105-112). Morristown, NJ: Association for Computational Linguistics.

Riloff, E., Wiebe, J., & Wilson, T. (2003). Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, May 27-June 1, 2003, Edmonton, Canada (pp. 25-32). Morristown, NJ: Association for Computational Linguistics.

Sebastiani, F. (2006). Classification of automatic text In K. Brown (Ed.), *The encyclopedia of language and linguistics* (2nd ed., Vol. 14, pp. 457-462). Amsterdam, NL: Elsevier Science Publishers.

Sifry, D. (2008). *State of the blogosphere 2008*. Retrieved May 12, 2009, from http://technorati.com/blogging/feature/state-of-the-blogosphere-2008/

Sifry, D. (2007). *The state of the live Web*. Retrieved September 12, 2008, from http://technorati.com/weblog/2007/04/328.html

Singer, J.L. and Salovey, P. (1988). Mood and memory: Evaluating the network theory of affect. *Clinical Psychology Review*, 8, 211-251.

Søgaard, A. (2010). Simple semi-supervised training of part-of-speech taggers. *In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, Uppsala, Sweden (pp. 205-208).

Stone, P. J. (1997). Thematic text analysis: New agendas for analyzing text content. In C. Roberts (Ed.), Text analysis for the social sciences. Mahwah NJ: Lawrence Erlbaum Associates.

Suzuki, Y., Takamura, H., & Okumura, M. (2006). Application of semi-supervised learning to evaluative expression classification. In *Seventh international conference on Computational linguistics and intelligent text processing (CICLING)*, (pp. 502-513).

Takamura, H., Inui, T., & Okumura, M. (2006). Latent variable models for semantic orientations of phrases. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, (pp. 201-208).

Tan, S., Cheng, X., Wang, Y., & Xu, H. (2009). Adapting Naïve Bayes to domain adaptation for sentiment analysis. In *Proceedings of the 31st European Conference on Information Retrieval (ECIR'2009)*, (pp. 337-349).

Thelen, M., & Riloff, E. (2002). A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *ACL-02 Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA (pp. 214-221).

Thomas, M., Pang, B., & Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, July 22-23, 2006, Sydney, Australia (pp. 327–335).

Tokuhisa, R., & Terashima, R. (2006). Relationship between utterances and enthusiasm in non-task-oriented conversational dialogue. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, July 15-16, 2006, Sydney, Australia (pp. 161-167). Morristown, NJ: Association for Computational Linguistics.

Tong, S., & Koller, D. (2001). Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2, 45-66.

Tsou, B. K. Y., Yuen, R. W. M., Kwong, O. Y., Lai, T. B. Y., & Wong, W. L. (2005). Polarity classification of celebrity coverage in the Chinese press. In *Proceedings of the International Conference on Intelligence Analysis*, May 2-4, 2005, McLean, VA.

Turney, P. D., & Littman, M. L. (2002). *Unsupervised learning of semantic orientation from a hundred-billion-word corpus* (Tech report No. ERB-1094): National Research Council Canada, Institute for Information Technology. (N. R. C. Canada No. Document Number)

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

Vechtomova, O. (2007). Using subjective adjectives in opinion retrieval from blogs. Proceedings of the Sixteenth Text REtrieval Conference (TREC 2007).

Wang, W., Huang, Z., & Harper, M. (2007). Semi-supervised learning for part-of-speech tagging of Mandarin transcribed speech. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hawaii (pp. 137-140).

Wang, W., & Zhou, Z. H. (2007). Analyzing co-training style algorithms. In *Proceedings of the 18th European Conference on Machine Learning*, (pp. 454-465).

Whitelaw, C., Garg, N., & Argamon, S. (2005a). Using appraisal groups for sentiment analysis. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, October 31-November 05, Bremen, Germany (pp. 625-631). New York, NY: ACM.

Whitelaw, C., Garg, N., & Argamon, S. (2005b). Using appraisal taxonomies for sentiment analysis. In *Proceedings of MCLC-05, the 2nd Midwest Computational Linguistic Colloquium (MCLC 2005)*, Columbus, Ohio.

Wiebe, J. (2000). Learning subjective adjectives from corpora. In *Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000)*, July 30-August 03, 2000, Austin, TX (pp. 735-740). Menlo Park, CA: AAAI Press.

Wiebe, J., Bruce, R., Bell, M., Martin, M., & Wilson, T. (2001). A corpus study of evaluative and speculative language. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*, September 01-02, 2001, Aalborg, Denmark (pp. 1-10). Morristown, NJ: Association for Computational Linguistics.

Wiebe, J., Bruce, R., & O'Hara, T. P. (1999). Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, June 20-26, 1999, College Park, MD (pp. 246-253). Morristown, NJ: Association for Computational Linguistics.

Wiebe, J., & Mihalcea, R. (2006). Word sense and subjectivity. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL*, July 17-18, 2006, Sydney, Australia (pp. 1065-1072). Morristown, NJ: Association for Computational Linguistics.

Wiebe, J., & Riloff, E. (2005). Creating subjective and objective sentence classifiers from unannotated texts. In *Proceedings of the 6th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2005).* , Feb 13-19, 2005, Mexico City, Mexico (pp. 486-497). Heidelberg, Berlin: Springer-Verlag.

Wiebe, J., & Wilson, T. (2002). Learning to disambiguate potentially subjective expressions. *Proceedings of the 6th Conference on Natural Language Learning, 20*, 1-7.

Wiebe, J., Wilson, T., & Bell, M. (2001). Identifying collocations for recognizing opinions. In *Proceedings of the ACL-01 Workshop on Collocation:*

Computational Extraction, Analysis, and Exploitation, Toulouse, France (pp. 24-31).

Wiebe, J., Wilson, T., Bruce, R., Bell, M., & Martin, M. (2004). Learning subjective language. *Computational Linguistics, 30*(3), 277-308.

Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation, 39*(2), 165-210.

Wilson, T., Pierce, D. R., & Wiebe, J. (2003). Identifying opinionated sentences. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Demonstrations*, Edmonton, Canada (pp. 33-34). Morristown, NJ: Association for Computational Linguistics.

Wilson, T., Wiebe, J., & Hwa, R. (2004). Just how mad are you? Finding strong and weak opinion clauses. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-2004)*, July 25-29, 2004, San Jose, CA (pp. 761–769). Menlo Park, CA: AAAI Press.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics* (pp. 189–196).

Yi, J., Nasukawa, T., Bunescu, R., & Niblack, W. (2003). Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM-2003)* November 19-22, 2003, Melbourne, FL (pp. 427-434). Washington, DC: IEEE Computer Society.

Yi, J., & Niblack, W. (2005). Sentiment mining in WebFountain. In *Proceedings of the 21st International Conference on Data Engineering (ICDE-2005)*, April 05-08, 2005, Tokyo, Japan (pp. 10731083). Washington, DC: IEEE Computer Society.

Yang, K., Yu, N., & Zhang, H. (2007). WIDIT in TREC-2007 Blog track: Combining lexicon-based methods to detect opinionated blogs. *Proceedings of the 16th Text REtrieval Conference (TREC 2007)*.

Yu, H., & Hatzivassiloglou, V. (2003). Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, 10*, 129-136.

Zhang, L., Barnden, J. A., Hendley, R. J., & Wallington, A. M. (2006). Exploitation in affect detection in open-ended improvisational text. In *Proceedings of Workshop on Sentiment and Subjectivity in Text*, July 22, 2006, Sydney, Australia (pp. 47-54). Morristown, NJ: Association for Computational Linguistics.

Zhang, W., & Yu, C. (2007). UIC at TREC 2007 Blog track. *Proceedings of the 16th Text REtrieval Conference (TREC 2007)*.

Zhang, W., Yu, C., & Meng, W. (2007). Opinion retrieval from blogs. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, Lisbon, Portugal (pp. 831-840). New York, NY: ACM.

Zhang, Z. (2004). Weakly-supervised relation classification for information extraction. In *Proceedings of the 13th Conference of Information and Knowledge Management*, Washington, DC (pp. 581-588).

Zhou, G., Joshi, H., & Bayrak, C. (2007). Topic categorization for relevancy and opinion detection. *Proceedings of the 16th Text REtrieval Conference (TREC 2007)*.

Zhou, Y., & Li, M. (2005a). Semi-supervised regression with co-training. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, Edinburgh, Scotland (pp. 908-913).

Zhou, Z.-H., & Li, M. (2005b). Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge and Data Engineering*, 17(11), 1529-1541.

Zhu, X. (2008). *Semi-supervised learning literature survey*: Department of Computer Sciences, University of Wisconsin, Madison. (Technical Report No. 1530)

Appendix A

Levin's Verb Class Terms Related to Opinion Expressions

(Sorted by descending occurrence)

mourn	like	dishearten	rage	complain	astound
fear	value	placate	disappoint	dispirit	throw
lament	shame	agonize	boast	irk	outrage
abhor	respect	disquiet	relish	comfort	affront
deplore	miss	trouble	discourage	entice	inspire
dread	cheer	wow	alarm	repent	moon
envy	rejoice	exasperate	hypnotize	devastate	pique
rue	appreciate	flabbergast	revitalize	pacify	care
dislike	obsess	cherish	disgrace	fascinate	cow
execrate	gladden	tickle	exalt	excite	sanction
despise	enjoy	mesmerize	terrify	humble	marvel
regret	savor	torment	stupefy	intrigue	flatter
hate	love	nauseate	grouse	displease	elate
loathe	grumble	chill	soothe	disillusion	bewitch
resent	adore	spook	amuse	stir	tease
detest	tolerate	hearten	captivate	embarass	scandalize
distrust	hurt	intoxicate	brag	perturb	abash
pity	sadden	surprise	intimidate	jollify	crush
enthuse	jar	daze	bore	distract	disturb
worry	treasure	wound	relax	confuse	affect
sicken	crab	venerate	prize	insult	disarm
grieve	tire	lull	exhaust	sober	disgruntle
thrill	weary	revere	engross	astonish	concern
delight	calm	tempt	support	shake	excuse
trust	scare	worship	object	bewilder	repulse
favor	numb	invigorate	depress	miff	embarrass
madden	arouse	assuage	ruffle	entrance	perplex
disdain	aggravate	mortify	content	esteem	entertain
gripe	boggle	please	nettle	faze	daunt
admire	jolt	enchant	dismay	irritate	threaten
engage	idolize	exult	frighten	startle	incense
puzzle	kvetch	solace	console	frustrate	terrorize
grouch	stand	move	baffle	fluster	interest
anger	believe	bother	shock	annoy	dazzle
fancy	rankle	stagger	agitate	plague	sting

exhilarate	amaze	felicitate	condescend	transgress
afflict	enrapture	provoke	fatigue	chasten
demoralize	vex	refresh	disparage	pardon
enliven	mollify	convince	revenge	deprecate
offend	encourage	demolish	damn	grovel
confound	infuriate	upset	torture	fine
discomfit	dissatisfy	repel	despair	bask
stun	transport	applaud	matter	pamper
relieve	embolden	overawe	beware	revile
enthrall	dumbfound	harass	aggrieve	groove
bug	gall	discombobulate	acclaim	luxuriate
cut	electrify	haunt	scorn	interrogate
disgust	enlighten	mystify	recover	defend
distress	awe	enrage	revel	reflect
try	horrify	abuse	menace	overweigh
revolt	gratify	upbraid	denounce	square
pain	rhapsodize	debase	grudge	conform
wonder	reward	recharge	influence	triumph
satisfy	beguile	gibe	degrade	grate
discompose	stimulate	rally	reproach	acquaint
uplift	charm	rejuvenate	divert	double-cross
chagrin	deject	criticize	ruminate	glorify
strike	peeve	contemplate	deify	toast
floor	disconcert	advocate	attract	deride
muddle	touch	interview	bear	punish
reassure	alienate	excoriate	beset	gush
tantalize	fret	vilify	exhort	disserve
impress	unsettle	bless	chide	meditate
spellbind	humiliate	transfix	clinch	victimize
appal	titillate	familiarize	lack	penalize
fume	overwhelm	wrong	trick	glory
unnerve	stump	thank	denigrate	invoke
appease	rile	betray	single_out	foil
forgive	galvanize	quench	condemn	inconvenience
antagonize	preoccupy	greet	involve	cringe

assail	fail	implicate	recompense	reprove
overpower	salivate	stultify	gloat	react
budge	swoon	impair	pester	major
fault	attack	molest	decieve	forget
recuperate	scold	blaspheme	compliment	spur_on
adulate	bewail	acclimate	befuddle	appall
ravish	accustom	curse	chastise	sorrow
salute	disbelieve	approve	fool	honor
mope	rebuke	accurse	extol	taunt
deceive	indulge	censure	lambaste	inflame
depend	persecute	imperil	allay	concentrate
avenge	ache	confront	belittle	
castigate	consent	innervate	hunger	
suffer	generalize	condone	doubt	
imprecate	pestle	disapprove	niggle	
eulogize	discipline	satiate	panic	
ensure	commend	anguish	assault	
enervate	laud	wallow	indict	
alleviate	praise	vindicate	habituate	
seethe	feel	mock	celebrate	
slander	mind	suffice	harm	
busy	calumniate	compensate	prefer	
instigate	hail	reprimand	oppress	
focus	repay	congratulate	welcome	
defame	attest	manipulate	violate	
unlearn	jade	acclimatize	backbite	
overrule	cry	propitiate	impeach	
fib	mitigate	incriminate	taste	
reaffirm	back	malign	decry	
rely	adapt	endanger	affirm	
prosecute	vaunt	remunerate	blame	
importune	hallucinate	occupy	ridicule	
preserve	dishonor	muse	rave	
weep	snub	appeal	dream	
bleed	palpate	infatuate	prejudice	

Appendix B

FrameNet Category Labels Related to Opinion Expressions

Adjectives

amazing	desirable	interested	sensational
ambsorbed	disapproving	irritated	shitty
admiring	disdainful	joyful	so-so
afraid	disparaging	jubilant	splendid
agape	dissatisfied	lame	stupendous
appalling	dreadful	laudable	substandard
appreciative	empathetic	laudatory	super
apprehensive	excellent	magnificent	superb
approving	execrable	maledictory	superlative
astonishing	execrative	marvellous	taken
astounding	fabulous	mediocre	terrible
average	fair	mocking	terrific
awful	fantastic	nervous	terrified
bad	fazed	nettled	third-rate
calm	fed up	okay	tip-top
commendable	fine	outstanding	tolerable
commendatory	first-rate	pathetic	top-notch
contemptuous	fond	pitiful	tremendous
crappy	freaked	reprehensible	uncool
critical	frightened	reproachful	uncritical
decent	fulfilled	rotten	unfazed
denigrative	good	rueful	upset
denunciative	great	satisfied	wonderful
denunciatory	grief-stricken	scared	worked up
deprecative	horrible	scathing	worried
deprecatory	incredible	scornful	wrapped up in
derisive	inferior	second-rate	

Nouns

abhorrence	conjecture	excellence	proposition
acclaim	contempt	exclamation	recrimination
accolade	contention	excoriation	refusal
accusation	critic	execration	regret
acknowledgment	criticism	explanation	relish
admiration	critique	fault	remark
admission	damnation	fear	remonstrance
adoration	deception	fib	report
affirmation	declaration	fulfillment	reprehension
allegation	denial	harangue	reproach
announcement	denigration	hatred	resentment
antipathy	denouncement	hunch	respect
appreciation	denunciation	insistence	reverence
approbation	deprecation	kudos	ridicule
assertion	derision	lamer	satisfaction
avowal	desperation	lie	scorn
belief	detestation	loathing	shocker
belittlement	disapproval	malediction	solace
belittling	disdain	mention	statement
blame	dislike	misrepresentation	stigma
censure	disparagement	mockery	stricture
charge	disrespect	opinion	surprise
claim	dread	pity	take
comfort	empathy	pleasure	vexation
commendation	enjoyment	praise	view
comment	envy	prevarication	vilification
compassion	equivocation	proclamation	
concession	esteem	pronouncement	
condemnation	exaltation	proposal	

Verbs

abash	decline	gratify	reiterate
abhor	decry	grieve	relate
abominate	deify	harangue	relish
acclaim	delight	hate	remark
accuse	denigrate	have feeling	remonstrate
acknowledge	denounce	hazard	repel
add	deny	hearten	report
address	deplore	hoodwink	reprehend
admire	deprecate	humiliate	resent
admit	depress	impress	respect
adore	deride	incense	revere
affirm	despair	infuriate	revolt
aggravate	despise	insist	ridicule
aggrieve	detest	interest	rile
agree	disappoint	intimidate	rock
alarm	disapprove	intrigue	rue
allege	discomfit	irk	sadden
amaze	disconcert	irritate	satisfy
anger	discourage	joke	savour
announce	disdain	kid	say
annoy	dishearten	laud	scare
antagonize	dislike	lie	scoff
applaud	disparage	like	scorn
appreciate	displease	loathe	see eye to eye
assert	distress	love	set store
astonish	disturb	luxuriate	shake
astound	dread	madden	shame
attest	dump	maintain	shit
aver	embarrass	mention	shock
avow	embitter	mislead	sicken
baffle	empathize	misrepresent	slam
beguile	enchant	mock	smirk
belittle	enjoy	mollify	sober
bewilder	enrage	mortify	solace
bewitch	entertain	mourn	soothe
blame	enthrall	mystify	speak
blast	envy	nettle	spook
boggle	equivocate	nonplus	startle
boo	esteem	note	state

bore	exalt	observe	stigmatize
bullshit	exasperate	offend	stimulate
calm	excite	outrage	sting
captivate	exclaim	pacify	stir
castigate	excoriate	perplex	stun
caution	execrate	perturb	stupefy
censure	exhilarate	petrify	suck
charge	expect	pity	suggest
charm	explain	placate	suppose
cheer	extol	please	surprise
cite	fascinate	pout	talk
claim	fault	praise	terrify
comfort	faze	preach	think
commend	fear	prevaricate	thrill
comment	feel	prize	tickle
concede	fib	proclaim	torment
conciliate	figure	profess	tout
condemn	flabbergast	propose	traumatize
confirm	floor	pull the wool over (someone's) eyes	trouble
confuse	flummox	pull-leg	unnerve
conjecture	fluster	puzzle	unsettle
console	fool	rankle	upset
contend	frighten	rattle	value
criticize	frustrate	reaffirm	venture
critique	fulfil	reassure	vex
damn	gall	recount	wow
dazzle	gibe	recriminate	write
deceive	gladden	refuse	
declare	gloat	regret	

Appendix C

IU Collocations

I abhor,abhorred, abhorring	I'm dismayed	my peeve
I accept	I'm dispassionate	my perception
I accuse	I'm displeased	my perspective
I acknowledge	I'm dissatisfied	my perturbation
I admire	I'm distraught	my pick
I admit	I'm distraut	my pleasure
I adore	I'm distrustful	my point
I advise	I'm doubtful	my position
I affirm	I'm drawn	my posit
I agree,agreed,agreeing	I'm dubious	my postulate
I allow	I'm dumbfounded	my postulation
I analyse	I'm dumbstricken	my praise
I analyze	I'm dumbstruck	my prediction
I anticipate	I'm dumfounded	my preference
I applaude	I'm eager	my pride
I applaud	I'm embarrassed	my process
I appraise	I'm emotional	my proposition
I appreciate	I'm enchanted	my puzzlement
I approve	I'm encouraged	my question
I approximate	I'm entertained	my ranting
I argue	I'm enthusiastic	my rating
I ascertain	I'm exasperated	my reaction
I assert	I'm excited	my realisation
I assess	I'm facinated	my realization
I assume	I'm fed up	my reasoning
I assure	I'm feverish	my reason
I attest	I'm flabbergasted	my reassurance
I believe	I'm foaming	my reckoning
I bet,bet,betting	I'm fond	my recommendation
I calculate	I'm frantic	my reflection
I care	I'm frenzied	my regret
I categorize	I'm frightened	my rejection
I challenge	I'm frothing	my remark
I cheer	I'm frowning	my renouncement
I cherish	I'm furious	my repudiation
I choose	I'm glad	my repulsion
I claim	I'm gloomy	my resentment

I classify,classified, classifying	I'm grateful	my resolve
I comment	I'm gratified	my respect
I commit,committed, committing	I'm grave	my response
I communicate	I'm groping	my reverance
I complain	I'm haggard	my reverence
I comprehend	I'm haunted	my reverie
I conceive	I'm heavyhearted	my review
I conclude	I'm hesitant	my rilement
I concur,concurred, concurrring	I'm horrified	my rumination
I condone	I'm hurt	my salutation
I confirm	I'm hypercritical	my sanction
I consent	I'm hysterical	my say
I consider	I'm ill	my score
I contemplate	I'm impressed	my scrutiny
I contend	I'm inclined	my selection
I contradict	I'm indifferent	my sense
I convey	I'm influenced	my sensing
I crave	I'm infused	my sensitivity
I credit	I'm interested	my sentiment
I cringe	I'm intoxicated	my shunning
I criticise	I'm intrigued	my speculatation
I criticize	I'm invigorated	my standing
I debate	I'm irritated	my statement
I decide	I'm jolted	my suggestion
I declare	I'm lost	my summary
I deduce	I'm lost	my support
I deduct	I'm melancholic	my supposition
I deem	I'm mesmerized	my surprise
I deny,denied,denying	I'm miffed	my suspicion
I deplore	I'm miserable	my sympathy
I desire	I'm mixed up	my synthesis
I despise	I'm mournful	my take
I detect	I'm mystified	my tale
I determine	I'm nauseated	my tally
I detest	I'm nettled	my taste
I differ	I'm nonplused	my thanks
I dig,dug,digging	I'm nonplussed	my theory
I disagree,disagreed, disagreeing	I'm offended	my thinking
I disapprove	I'm openmouthed	my thought

I disavow	I'm opposed	my trust
I disbelieve	I'm optimistic	my turnoff
I discover	I'm overcome	my turnon
I discuss	I'm overcritical	my unbelief
I dislike	I'm overwrought	my uncertainty
I dissent	I'm pained	my understanding
I distrust	I'm partial	my utterance
I doubt	I'm passionate	my verification
I dream	I'm peeved	my vexation
I empathize	I'm pensive	my view
I emphasise	I'm perplexed	my vision
I emphasize	I'm persuaded	my vomiting
I enjoin	I'm perturbed	my vomit
I enjoy	I'm pessimistic	my vote
I entertain	I'm pissed	my want
I envisage	I'm pleased	my warning
I esteem	I'm popeyed	my wisdom
I estimate	I'm positive	my wish
I evaluate	I'm proud	my wonderment
I examine	I'm puzzled	my wonder
I exclaim	I'm ragged	my word
I expect	I'm reassured	my worry
I explain	I'm refreshed	my worship
I express	I'm reinvigorated	my yearning
I faith	I'm relieved	as for me
I fancy,fancied,fancying	I'm repulsed	bad for me
I fantasize	I'm repulsed	enough for me
I favor	I'm reverent	good for me
I fear	I'm riled	okay for me
I feel,felt,feeling	I'm riled	accommodate me
I figure	I'm rivetted	affect me
I find	I'm ruffled	aggravate me
I focus	I'm saddened	agitate me
I gather	I'm sad	agree with me
I gauge	I'm satisfied	agrrevate me
I get	I'm seething	alarm me
I glean	I'm serious	alienate me
I grant	I'm shaken	allow me
I grasp	I'm sickened	amaze me
I guess	I'm sick of	amuse me
I hate	I'm smitten	annoy me
I hazard	I'm sold	appalling to me

I hypothesize	I'm soothed	appeal to me
I identify	I'm startled	appear to me
I imagine	I'm steamed	arouse me
I imply	I'm stirred	assail me
I indicate	I'm stuck	assuage me
I induce	I'm stung	assure me
I infer	I'm stunned	astonish me
I inform	I'm stupefied	astound me
I insist	I'm suited	attract me
I interpret	I'm supercritical	baffle me
I intuit	I'm sure	bear upon me
I judge	I'm sure	become me
I justify	I'm surprised	bedazzle me
I know	I'm suspicious	bedevil me
I label	I'm thankful	befuddle me
I lean	I'm thunderstruck	beguile me
I like	I'm tired of	believe me
I loathe	I'm troubled	bewilder me
I long	I'm unbiased	bewitch me
I love	I'm uncertain	blind me
I maintain	I'm uncomfortable	bore me
I marvel	I'm uneasy	bother me
I measure	I'm unsatisfied	bound me
I mind	I'm unsure	break me
I miss	I'm upset	bring,brought,bringing me
I mistrust	I'm vexed	calm me
I mull	I'm weary	captivate me
I muse	I'm worried	capture me
I note	my abohrance	catch,caught,catching me
I notice	my acceptance	chafe me
I observe	my accounting	challenge me
I okay	my account	change me
I oppose	my acknowledgement	charm me
I pass	my acknowledgment	cheer me
I perceive	my admiration	cheer me
I pick	my admission	churn me
I pity	my adoration	command me
I point out	my advice	compel,compelled,compelling me
I ponder	my affection	concern me
I pose	my affinity	confine me
I posit	my agitation	confound me

I postulate	my aim	confuse me
I praise	my alienation	consume me
I predict	my amazement	contain me
I prefer	my ambivalence	control me
I present	my amusement	convert me
I presume	my analysis	convey me
I presuppose	my anger	convince me
I pretend	my anguish	correct me
I prise	my annoyance	cover me
I prize	my answer	cross me
I proclaim	my antagonism	damage me
I pronounce	my anxiety	dazzle me
I propose	my appraisal	defeat me
I question	my appreciation	delight me
I raise	my apprehension	deliver me
I rant	my approval	demand me
I rate	my approving	detain me
I rave	my approximation	deter me
I react	my argument	direct me
I realise	my assent	discomfit me
I realize	my assertion	disconcert me
I reason	my assessment	disgust me
I reassure	my assumption	displease me
I rebut	my assurance	disquiet me
I reckon	my astonishment	dissatisfy,dissatisfied,dissatisfying me
I recognise	my attestment	dissuade me
I recognize	my attitude	distract me
I recommend	my attraction	disturb me
I reflect	my aversion	draw,drew,drawing me
I refuse	my awareness	draw me
I refute	my awe	drive,drove,driving me
I regard	my bafflement	elicit me
I regret	my bedazzlement	elucidate me
I reiterate	my beef	enamor me
I reject	my belief	enamour me
I rejoice	my bet	enchant me
I relate	my bewilderment	encourage me
I rely	my blessing	engage me
I remain	my calculation	enlighten me
I remark	my capitulation	ensure me
I renounce	my categorisation	entertain me
I repudiate	my categorization	evoke me

I resent	my certainty	exacerbate me
I respect	my chafe	exasperate me
I respond	my choice	excite me
I restate	my claim	fascinate me
I retell	my classification	find me
I revere	my comfort	fit me
I review	my commendation	fix me
I risk	my comment	floor me
I root	my commitment	flush me
I ruminate	my communicating	force me
I sanction	my communication	frighten me
I savor	my complaint	get to me
I savour	my computation	give me
I savvy	my concept	gladden me
I say,said,saying	my conceptualisation	grab,grabbed,grabbing me
I see	my conceptualization	gratify,gratified, gratifying me
I select	my concern	gross out me
I sense	my conclusion	grow on me
I speak	my confidence	haunt me
I speculate	my conflict	have,had,having me
I state	my confusion	hit,hit,hitting me
I stress	my congratulation	hold,held,holding me
I struggle	my conjecture	horrify,horrified, horrifying me
I suffer	my consent	hurt me
I suggest	my consideration	ignite in me
I suppose	my contempt	impact me
I surmise	my contention	impel me
I suspect	my contentment	impress me
I swear	my counterpoint	impression on me
I sympathize	my craving	incapacitate me
I take	my criticism	incise me
I talk	my criticism	induce me
I tally	my critique	influence me
I tell,told,telling	my curiosity	infuse me
I thank	my decision	injure me
I think,thought,thinking	my delight	inspire me
I treasure	my desire	interest me
I treat	my despair	intimidate me
I trust	my diffidence	intrigue me
I	my disagreement	invigorate me

understand,understood,understanding		
I undervalue	my disapproval	involve me
I utter	my disbelief	irritate me
I value	my discomfort	jazz me
I venture	my discontent	jolt me
I verbalise	my discovery	jolt me
I verbalize	my discussion	keep,kept,keeping me
I verify	my disdain	kill me
I view	my disgust	leave,left,leving me
I vote	my dislike	look to me
I vouch	my dismay	make,made,making me
I want	my disparagement	move me
I warn	my displeasure	mystify,mystified,mystifying me
I wish	my disposition	nauseate me
I wonder	my dispute	nettle me
I worry	my disrespect	occur,occurred,occurring to me
I yearn	my dissatisfaction	offend me
I'm acritical	my dissent	offer me
I'm affected	my distress	pain me
I'm afraid	my distrustfulness	permit me
I'm aghast	my distrust	persuade me
I'm agitated	my doubtfulness	perturbe me
I'm alarmed	my doubt	perturb me
I'm alienated	my dream	piss me
I'm amazed	my dubiousness	please me
I'm ambivalent	my emotion	prepare me
I'm amused	my emphasis	present me
I'm angry	my enjoyment	pressure me
I'm annoyed	my enmity	prevent me
I'm antagonistic	my esteem	propel me
I'm anti	my estimate	protect me
I'm anxious	my estimation	provide me
I'm appalled	my evaluation	provoke me
I'm appreciative	my examination	push me
I'm apprehensive	my expectation	put off me
I'm aroused	my explaination	puzzle me
I'm ashamed	my explanation	quieten me
I'm assailed	my expression	quiet me
I'm assuaged	my eyes	rag me
I'm assured	my facination	raise me

I'm astonied	my faith	reassure me
I'm astonished	my fantasy	reinvigorate me
I'm astounded	my favorite	release me
I'm attracted	my fear	relieve me
I'm aware	my feeling	render me
I'm aweary	my figuring	repel me
I'm awed	my focus	repulse me
I'm awestruck	my fondness	restrain me
I'm baffled	my foreboding	restrict me
I'm beat	my fright	rile me
I'm bedazzled	my grasp	rouse me
I'm befuddled	my gratitude	rub me
I'm beguiled	my gripe	ruffle me
I'm bemused	my guess	sadden me
I'm bewildered	my hate	satisfy,satisfied, satisfying me
I'm bewitched	my hatred	scare me
I'm biased	my head	screw me
I'm bittersweet	my heart	secure me
I'm blearyeyed	my hopefulness	seem to me
I'm bleary	my hostility	seize me
I'm blinded	my hunch	sell,sold,selling me
I'm blistering	my hurting	sense to me
I'm bored	my hurt	set back me
I'm bothered	my hypothesis	shag,shagged,shagging me
I'm bound	my idea	shake,shook,shaking me
I'm broken	my imagination	shake up me
I'm bubbling	my impression	sicken me
I'm burndout	my inclination	soothe me
I'm burned	my indecision	sour me
I'm bushed	my indecisiveness	spread over me
I'm calmed	my insight	squeeze me
I'm captivated	my interest	stagnate me
I'm captured	my interpretation	startle me
I'm carping	my intuition	stick me
I'm censorious	my intution	stimulate me
I'm certain	my ire	sting me
I'm certain	my irritation	stir me
I'm chafed	my issue	stir me
I'm chagrined	my joy	stop me
I'm cheered	my judgement	straighten out me
I'm cheerful	my judgment	strain me

I'm chuffed	my justification	strike,struck,striking me
I'm churning	my knowing	stupefy,stupefied,stupefying me
I'm cognizant	my knowingness	suggest to me
I'm comforted	my knowledge	suit me
I'm compelled	my like	surprise me
I'm concerned	my liking	sustain me
I'm confident	my loathing	teach me
I'm conflicted	my logic	tell,told,telling me
I'm confounded	my longing	threaten me
I'm confused	my love	throttle me
I'm consumed	my loyalty	throw me
I'm contented	my meaning	touch me
I'm content	my measurement	transform me
I'm convinced	my message	trouble me
I'm crazy about	my mind	trust me
I'm critical	my mistrust	turn me
I'm delighted	my mood	unhinge me
I'm delivered	my mulling	upset,upsetted,upsetting me
I'm deterred	my need	vex me
I'm discomfited	my notion	violate me
I'm discontented	my objection	warn me
I'm discontent	my observation	win,won,winning me
I'm discriminative	my opinion	work me
I'm disgusted	my opposition	worry,worried,worrying me
I'm disinterested	my orientation	

Appendix D

Review Bigrams

! a	excellent !	recommend it
! great	fabulous !	recommend this
! if	fantastic !	recommended !
! it	first time	right !
! the	for and	rocks !
! there	for anyone	simply the
! you	forward to	so much
?a really	full of	soon as
?for us	good !	superb !
a fantastic	great little	surprisingly good
a perfect	had never	thank you
a real	have always	thanks to
a terrific	have ever	the only
a winner	have never	the real
a wonderful	highly recommend	the very
absolutely no	highly recommended	there !
all you	i can't	time !
an absolute	i could	to anyone
an amazing	i ever	to help
an awesome	i hope	top of
an extremely	i never	up !
and just	i recommend	up the
and never	i say	us !
and yes	i will	very enjoyable
anyone who	i've ever	very helpful
are so	in such	very much
as possible	is absolutely	very very
as this	is all	was as
back !	it !	way !
believe it	just like	we could
best ever	kids !	we love
best of	like to	we will
brilliant !	love it	we would
by far	love the	well worth
can i	love this	what a
can't wait	loved this	what an
care of	managed to	will never
day !	me !	winner !

do anything	money !	wish i
don't be	my first	without a
don't let	my heart	wonderful !
enjoyed this	my own	world !
especially for	not only	worth every
even a	of all	worth it
even better	once again	wow !
even for	only thing	you !
even more	our own	you can't
even the	out !	you must
ever !	outstanding !	you see
every penny	people who	

Appendix E

Stop Word List

a	gets	look	seeing	until
an	getting	looking	self	unto
and	gives	looks	selves	up
are	go	ltd	sent	upon
at	goes	m	seven	us
b	going	many	several	use
be	gone	may	she	used
by	got	maybe	six	uses
c	gotten	meanwhile	some	using
com	h	much	somewhere	usually
come	had	must	sub	uucp
comes	happens	n	sup	v
contain	has	name	t	#very
containing	have	nd	take	via
contains	having	near	th	viz
course	he	need	that	w
d	hello	needs	thats	was
did	her	next	the	way
do	here	nine	their	we
does	hereafter	now	theirs	went
doing	herself	nowhere	them	were
done	hi	o	themselves	what
during	him	of	then	when
e	himself	off	thence	whence
each	his	oh	there	whenever
edu	hither	on	thereafter	where
eg	how	once	thereby	whereafter
eight	i	one	these	whereby
et	ie	ones	they	wherein
etc	if	onto	third	whereupon
every	in	or	this	wherever
everybody	inc	other	those	which
everyone	inner	others	three	while
everything	insofar	otherwise	through	whither
everywhere	into	out	throughout	who
ex	inward	outside	thru	whoever
f	is	over	to	whole
few	it	overall	together	whom

fifth	its	p	too	whose
five	itself	per	took	will
followed	j	placed	toward	with
following	k	provides	towards	within
follows	keep	q	tried	x
for	keeps	que	tries	y
former	kept	qv	try	you
formerly	l	r	trying	your
forth	lately	rd	twice	yours
four	later	re	two	yourself
from	latter	s	u	yourselves
g	latterly	second	un	z
get	let	secondly	under	zero

Ning Yu

School of Library and Information Science
Indiana University Bloomington
1320 E. 10th St., LI
Bloomington, IN 47405

http://ella.slis.indiana.edu/~nyu
Office: 812-856-5874
Cell: 812-325-2045
E-mail: nyu@indiana.edu

EDUCATION

Ph.D. in Information Science, April 2011 (expected)

Minor in Computational Linguistics and Cognitive Science

Indiana University, Bloomington IN

Dissertation: "Semi-Supervised Learning for Opinion Detection in the Blogosphere"

Master of Information Science, 2004, Indiana University, Bloomington IN

Bachelor of Information Management, 2001, Shandong University, China

RESEARCH EXPERIENCE

Opinion Detection

Ongoing research on automatic identification of opinions in user generated content (e.g., blogs). Immediate applications include opinion search, automatic tagging, and product/service monitoring:

- Designed and developed an opinion retrieval system that yielded several top runs for the opinion retrieval task in Blog track 2006 and 2007.

- Developed a prototype for identifying user intentions and valuable recommendation information from instant messaging (IM) data, in collaboration with Yahoo! Research.
- Currently exploring semi-supervised learning methods for tackling major challenges in opinion detection (i.e., insufficient labeled data and domain transfer).

Knowledge Management in Digital Libraries

Contributed to the management of digital library resources by developing and assigning metadata schema using a semi-automatic approach that leveraged both human expertise and machine techniques for the Just-in-Time Teaching Digital Library (JiTTDL), an NSF-funded project at the US Air Force Academy:

- Extracted and grouped concept words for creating a faceted classification for the Just-in-Time pedagogical resources.
- Designed a digital library portal with coarse-to-fine search options, including a prototype for faceted retrieval.

Information Retrieval (IR) & Data Mining

Researched the development of fusion-based IR systems by identifying various sources of evidence using both machine learning and natural language processing techniques:

- Participated in multiple retrieval tasks (e.g., spam identification, blog retrieval) for TREC (Text REtrieval Conference), where the research team achieved top performance.
- Designed strategies for Web crawling, query expansion (e.g., identifying noun phrase) and re-ranking (e.g., optimizing parameters for sources of evidence).
- Contributed to the development and evaluation of the Web Information Discovery Integrated Tool (WIDIT), an environment for prototyping IR systems and fostering collaborative research.

Information Visualization

Applied visualization techniques to large-scale data to reveal latent knowledge:

- Conducted time-series analyses on email transactions to reveal interaction behavior between reference librarians and students.

- Generated static and dynamic visualizations to identify interlinked clusters and demonstrate the distribution, evolution and connections of bloggers in the blogosphere.
- Presented weblog visualization at Language of Networks, Ars Electronica 2004, Austria.

PROFESSIONAL EXPERIENCE

Adjunct Lecturer 07/2010 - 12/2010

School of Library and Information Science, Indiana University Bloomington

S511 Database Design. Graduate course designed to teach theories and principles as well as practical design and implementation skills to create user-centered database management applications.

S603 Workshop in Library and Information Science: Perl/CGI. Graduate course in programming for information management designed to teach basic skills for creating dynamic and interactive Web applications using Perl and CGI.

Guest Lecturer Spring 2007, 2009

School of Library and Information Science, Indiana University Bloomington

S534 Information Retrieval: Theory and Practice. Graduate course that introduces basic IR theory and examines cutting-edge IR research in the context of TREC:

- Led general discussions regarding IR, TREC, and WIDIT.
- Presented lectures on opinion detection theory and practice.

Research Intern 05/2008 - 08/2008

Yahoo! Research, Santa Clara, CA

Conducted preliminary research on conversational mining in order to understand how people use instant messaging (IM) to coordinate, schedule tasks, and share recommendations with each other as part of a joint effort between Yahoo!'s Internet Experiences and Metrics Analysis groups:

- Examined the nature of conversational data to determine the feasibility of applying state-of-the-art opinion mining approaches to online conversations.
- Designed and conducted a field study to collect IM data.

- Performed primary statistical analysis on both IM data and chatroom data from Live.Yahoo.com and identified IM specific lexicon.

System Developer 09/2007 - 04/2008

IU Digital Library Program, Indiana University Bloomington

Provided programming and software support for a variety of Digital Library projects using various technologies (e.g., Perl, Java, JSP, XSLT, XML, JavaScript, CSS and HTML):

- Deployed and customized DSpace (a JAVA-based platform) to create an online repository for preserving and disseminating IU scholarly work.
- Automated the metadata mapping process.

Research Assistant 08/2005 - 08/2007

School of Library and Information Science, Indiana University Bloomington

Assisted faculty member with research projects using various technologies (e.g., Perl, Python, and Java):

- Led small teams of 2-4 developers in collaborating with senior IR researcher.
- Worked on several data mining projects in the field of curriculum analysis, U.S. pattern analysis and computer-mediated communication.

Teaching Assistant 09/2004 - 04/2007

School of Library and Information Science, Indiana University Bloomington

S517 Web Programming.

- Tutored lab sessions and graded homework.
- Presented 1-2 lectures each semester as a substitute instructor.

Graduate Assistant 08/2004 - 08/2005

School of Health, Physical Education & Recreation, Indiana University Bloomington

Served as webmaster and project coordinator:

- Advised faculties and graduate assistants on best Web accessibility.

- Designed and deployed online survey interfaces and databases, including a national 3-phrase survey which involved more than 4000 participants and hundreds of questions.

Reference Assistant 01/2003 - 08/2004

Walden University Virtual Library

- Answered reference questions via email and phone.
- Led summer workshop on how to search academic databases for thesis writing.

System Designer 07/2001 - 06/2002

DaZhong Publishing Information Technology, Jinan, Shandong, China

- Responsible for meeting with clients and analyzing information need.
- Designed information system prototype.

PUBLICATIONS

1. Yu, N. & Kübler, S. (2010). Semi-Supervised Learning for Opinion Detection. *The 2010 IEEE/WIC/ACM International Conferences on Web Intelligence (WI'2010))*, IEEE Computer Society Press. pdf
2. Yang, K., Yu, N., & Zhang, H. (2008). WIDIT in TREC2007 Blog track: Combining lexicon-based methods to detect opinionated blogs. *Proceedings of the 16th Text Retrieval Conference*. pdf
3. Yang, K., Yu, N., Valerio, A., Zhang, H., & Ke, W. (2007). Mining for opinions in blogosphere. Accepted as a regular paper at the Industry Track Workshop in the *11th Pacific-Asia Conference on Knowledge Discovery and Data Mining*.
4. Yang, K., Yu, N., Valerio, A., Zhang, H., & Ke, W. (2007). WIDIT in TREC2006 Blog track. *Proceedings of the 15th Text Retrieval Conference* (TREC2006). pdf
5. Yang, K., Yu, N., Valerio, A., Zhang, H., & Ke, W. (2007). Fusion approach to finding opinions in blogosphere. *Proceedings of the International Conference on Weblogs and Social Media.* **Nominated for Best Paper Award**. pdf
6. Lee, S., Jacob, E.K., Loehrlein, A., Yang, K., and Yu, N. (2006) Semi-automatic construction of a faceted scheme for knowledge discovery on the Web. *IFLA Conference*. August 20-24, 2006, Seoul, South Korea. (Poster Presentation)

7. Yang, K., Yu, N., Zhang, H., Akram, S., & Record, I. (2006). WIDIT: Integrated approach to HARD topic search. *Proceedings of 2006 Asian Information Retrieval Symposium*. pdf
8. Yang, K., Yu, N. (2005). WIDIT: Fusion-based approach to Web search optimization. *Proceedings of Information Retrieval Technology, Second Asia Information Retrieval Symposium* (AIRS 2005). pdf
9. Yang, K., Yu, N., George, N., Loehrlen, A., McCaulay, D., Zhang, H., Akram, S., Mei, J., Record, I. (2006). WIDIT in TREC 2005 HARD, Robust, and SPAM tracks. *Proceedings of the Fourteenth Text REtrieval Conference (TREC 2005)*, Gaithersburg, MD, USA, 2005. pdf
10. Yu, N. (2005) Where IR and IO meet in digital libraries: A fusion approach for knowledge representation and discovery. Abstract accepted by *Connections 2005: The 10th Great Lakes Information Science Conference*.
11. Loehrlein, A., Jacob, E. K., Yang, K., Lee, S., & Yu, N. (2005). A hybrid approach to faceted classification based on analysis of descriptor suffixes. *Proceedings of 2005 Annual Meeting of the American Society for Information Science and Technology.* pdf
12. Yang, K., Yu, N. & Lee, Yoon. (2005). Dynamic tuning for fusion: Harnessing human intelligence to optimize system performance. *Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics.* pdf
13. Herring, S. C., Kouper, I., Paolillo, J. C., Lois Ann Scheidt, Tyworth, M., Welsch, P., Wright, E., & Yu, N. (2005). Conversations in the blogosphere: An analysis "from the bottom up", *Proceedings of the Thirty-Eighth Hawaii International Conference on System Sciences (HICSS-38)*. Los Alamitos: IEEE Press. **Nominated for Best Paper Award**. pdf
14. Yang, K., Yu, N., Wead, A., Rowe, G. L., Li, Y.-H., Friend, C., et al. (2004). WIDIT in TREC-2004 Genomics, HARD, Robust, and Web tracks. *Proceedings of the 13th Text Retrieval Conference* (TREC2004). pdf
15. Yang, K., Jacob, E., Loehrlein, A., Lee, S., & Yu, N. (2004). The best of both worlds: A hybrid approach to the construction of faceted vocabularies. *Proceedings of Information Retrieval Technology, First Asia Information Retrieval Symposium (AIRS 2004).* pdf
16. Yang, K., Jacob, E., Loehrlein, A., Lee, S., & Yu, N. (2004). Organizing the web: Semi-automatic construction of a faceted scheme. *Proceedings of IADIS WWW/Internet 2004 Conference*, Madrid, Spain. pdf

AWARDS & HONORS

Indiana University Bloomington

- Best Presentation Award, 3rd Place, Doctoral Student Research Forum, 2010
- Travel Grant, 2010, 2009, 2007, 2005, 2003

- M I Rufsvold Fellowship, 2009
- Sarah Reed Fellowship, 2009
- Margaret Griffin Coffin Scholarship, 2008
- Merit Scholarship, 2004
- Student showcase award in Making IT Happen!, 2004

Shandong University, China

- First Prize Scholarship (top 3 students), 1998 - 2001
- San Zhu Fellowship (top 1 student), 1999
- Excellent Student leader (Vice-minister of the Public Relationship Department, Student Association), 1999

SERVICE

Journal reviewer:

- Scientometrics
- Journal of the American Society for Information Science and Technology

Conference reviewer:

- Instructional Systems Technology Conference 02/26, 2010, Indiana, US.
- Conference on Intelligent Text Processing and Computational Linguistics, 02/20-26, 2011, Tokyo, Japan

Other reviewer:

- Indiana Aspirations in Computing Award 2010/2011
- National Aspirations in Computing Award 2010/2011