



Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

A survey on sentiment detection of reviews

Huifeng Tang, Songbo Tan *, Xueqi Cheng

Information Security Center, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, PR China

ARTICLE INFO

Keywords:

Sentiment detection
Opinion extraction
Sentiment classification

ABSTRACT

The sentiment detection of texts has been witnessed a booming interest in recent years, due to the increased availability of online reviews in digital form and the ensuing need to organize them. Till to now, there are mainly four different problems predominating in this research community, namely, subjectivity classification, word sentiment classification, document sentiment classification and opinion extraction. In fact, there are inherent relations between them. Subjectivity classification can prevent the sentiment classifier from considering irrelevant or even potentially misleading text. Document sentiment classification and opinion extraction have often involved word sentiment classification techniques. This survey discusses related issues and main approaches to these problems.

© 2009 Published by Elsevier Ltd.

1. Introduction

Today, very large amount of reviews are available on the web, as well as the weblogs are fast-growing in blogosphere. Product reviews exist in a variety of forms on the web: sites dedicated to a specific type of product (such as *digital camera*), sites for newspapers and magazines that may feature reviews (like *Rolling Stone* or *Consumer Reports*), sites that couple reviews with commerce (like *Amazon*), and sites that specialize in collecting professional or user reviews in a variety of areas (like *Rottentomates.com*). Less formal reviews are available on discussion boards and mailing list archives, as well as in Usenet via Google Groups. Users also comment on products in their personal web sites and blogs, which are then aggregated by sites such as *Blogstreet.com*, *AllConsuming.net*, and *onfocus.com*.

The information mentioned above is a rich and useful source for marketing intelligence, social psychologists, and others interested in extracting and mining opinions, views, moods, and attitudes. For example, whether a product review is positive or negative; what are the moods among Bloggers at that time; how the public reflect towards this political affair, etc.

To achieve this goal, a core and essential job is to detect subjective information contained in texts, include viewpoint, fancy, attitude, sensibility etc. This is so-called *sentiment detection*.

A challenging aspect of this task seems to distinguish it from traditional topic-based detection (classification) is that while topics are often identifiable by keywords alone, sentiment can be expressed in a much subtle manner. For example, the sentence "What a bad picture quality that digital camera has! ... Oh, this

new type camera has a good picture, long battery life and beautiful appearance!" compares a negative experience of one product with a positive experience of another product. It is difficult to separate out the core assessment that should actually be correlated with the document. Thus, sentiment seems to require more understanding than the usual topic-based classification.

Sentiment detection dates back to the late 1990s (Argamon, Koppel, & Avneri, 1998; Kessler, Nunberg, & SchÄutze, 1997; Spertus, 1997), but only in the early 2000s did it become a major sub-field of the information management discipline (Chaovalit & Zhou, 2005; Dimitrova, Finn, Kushmerick, & Smyth, 2002; Durbin, Neal Richter, & Warner, 2003; Efron, 2004; Gamon, 2004; Gance, Hurst, & Tomokiyo, 2004; Grefenstette, Qu, Shanahan, & Evans, 2004; Hillard, Ostendorf, & Shriberg, 2003; Inkpen, Feiguina, & Hirst, 2004; Kobayashi, Inui, & Inui, 2001; Liu, Lieberman, & Selker, 2003; Raubert & Muller-Kogler, 2001; Riloff and Wiebe, 2003; Subasic & Huettner, 2001; Tong, 2001; Vegnaduzzo, 2004; Wiebe & Riloff, 2005; Wilson, Wiebe, & Hoffmann, 2005). Until the early 2000s, the two main popular approaches to sentiment detection, especially in the real-world applications, were based on machine learning techniques and based on semantic analysis techniques. After that, the shallow nature language processing techniques were widely used in this area, especially in the document sentiment detection. Current-day sentiment detection is thus a discipline at the crossroads of NLP and IR, and as such it shares a number of characteristics with other tasks such as information extraction and text-mining.

Although several international conferences have devoted special issues to this topic, such as ACL, AACL, WWW, EMNLP, CIKM etc., there are no systematic treatments of the subject: there are neither textbooks nor journals entirely devoted to sentiment detection yet.

* Corresponding author.

E-mail addresses: tanghuifeng@software.ict.ac.cn (H. Tang), tansongbo@software.ict.ac.cn (S. Tan), cxq@ict.ac.cn (X. Cheng).

This paper first introduces the definitions of several problems that pertain to sentiment detection. Then we present some applications of sentiment detection. Section 4 discusses the subjectivity classification problem. Section 5 introduces semantic orientation method. The sixth section examines the effectiveness of applying machine learning techniques to document sentiment classification. The seventh section discusses opinion extraction problem. The eighth part talks about evaluation of sentiment detection. Last section concludes with challenges and discussion of future work.

2. Sentiment detection

2.1. Subjectivity classification

Subjectivity in natural language refers to aspects of language used to express opinions and evaluations (Wiebe, 1994). Subjectivity classification is stated as follows: Let $S = \{s_1, \dots, s_n\}$ be a set of sentences in document D . The problem of subjectivity classification is to distinguish sentences used to present opinions and other forms of subjectivity (subjective sentences set S_s) from sentences used to objectively present factual information (objective sentences set S_o), where $S_s \cup S_o = S$. This task is especially relevant for news reporting and Internet forums, in which opinions of various agents are expressed.

2.2. Sentiment classification

Sentiment classification includes two kinds of classification forms, i.e., binary sentiment classification and multi-class sentiment classification. Given a document set $D = \{d_1, \dots, d_n\}$, and a pre-defined categories set $C = \{\text{positive, negative}\}$, binary sentiment classification is to classify each d_i in D , with a label expressed in C . If we set $C' = \{\text{strong positive, positive, neutral, negative, strong negative}\}$ and classify each d_i in D with a label in C' , the problem changes to multi-class sentiment classification.

Most prior work on learning to identify sentiment has focused on the binary distinction of positive vs. negative. But it is often helpful to have more information than this binary distinction provides, especially if one is ranking items by recommendation or comparing several reviewers' opinions. Koppel and Schler (2005a, 2005b) show that it is crucial to use neutral examples in learning polarity for a variety of reasons. Learning from negative and positive examples alone will not permit accurate classification of neutral examples. Moreover, the use of neutral training examples in learning facilitates better distinction between positive and negative examples.

3. Applications of sentiment detection

In this section, we will expound some rising applications of sentiment detection.

3.1. Products comparison

It is a common practice for online merchants to ask their customers to review the products that they have purchased. With more and more people using the Web to express opinions, the number of reviews that a product receives grows rapidly. Most of the researches about these reviews were focused on automatically classifying the products into "recommended" or "not recommended" (Pang, Lee, & Vaithyanathan, 2002; Ranjan Das & Chen, 2001; Terveen, Hill, Amento, McDonald, & Creter, 1997). But every product has several features, in which maybe only part of them people are interested. Moreover, a product has shortcomings in one aspect, probably has merits in another place (Morinaga,

Yamanishi, Tateishi, & Fukushima, 2002; Taboada, Gillies, & McFetridge, 2006).

To analysis the online reviews and bring forward a visual manner to compare consumers' opinions of different products, i.e., merely with a single glance the user can clearly see the advantages and weaknesses of each product in the minds of consumers. For a potential customer, he/she can see a visual side-by-side and feature-by-feature comparison of consumer opinions on these products, which helps him/her to decide which product to buy. For a product manufacturer, the comparison enables it to easily gather marketing intelligence and product benchmarking information.

Liu, Hu, and Cheng (2005) proposed a novel framework for analyzing and comparing consumer opinions of competing products. A prototype system called *Opinion Observer* is implemented. To enable the visualization, two tasks were performed: (1) Identifying product features that customers have expressed their opinions on, based on language pattern mining techniques. Such features form the basis for the comparison. (2) For each feature, identifying whether the opinion from each reviewer is positive or negative, if any.

Different users can visualize and compare opinions of different products using a user interface. The user simply chooses the products that he/she wishes to compare and the system then retrieves the analyzed results of these products and displays them in the interface.

3.2. Opinion summarization

The number of online reviews that a product receives grows rapidly, especially for some popular products. Furthermore, many reviews are long and have only a few sentences containing opinions on the product. This makes it hard for a potential customer to read them to make an informed decision on whether to purchase the product. The large number of reviews also makes it hard for product manufacturers to keep track of customer opinions of their products because many merchant sites may sell their products, and the manufacturer may produce many kinds of products.

Opinion summarization (Ku, Lee, Wu, & Chen, 2005; Philip et al., 2004) summarizes opinions of articles by telling sentiment polarities, degree and the correlated events. With opinion summarization, a customer can easily see how the existing customers feel about a product, and the product manufacturer can get the reason why different stands people like it or what they complain about.

Hu and Liu (2004a, 2004b) conduct a work like that: Given a set of customer reviews of a particular product, the task involves three subtasks: (1) identifying features of the product that customers have expressed their opinions on (called product features); (2) for each feature, identifying review sentences that give positive or negative opinions; and (3) producing a summary using the discovered information.

Ku, Liang, and Chen (2006) investigated both news and web blog articles. In their research, TREC, NTCIR and articles collected from web blogs serve as the information sources for opinion extraction. Documents related to the issue of animal cloning are selected as the experimental materials. Algorithms for opinion extraction at word, sentence and document level are proposed. The issue of relevant sentence selection is discussed, and then topical and opinionated information are summarized. Opinion summarizations are visualized by representative sentences. Finally, an opinionated curve showing supportive and non-supportive degree along the timeline is illustrated by an opinion tracking system.

3.3. Opinion reason mining

In opinion analysis area, finding the polarity of opinions or aggregating and quantifying degree assessment of opinions

scattered throughout web pages is not enough. We can do more critical part of in-depth opinion assessment, such as finding reasons in opinion-bearing texts. For example, in film reviews, information such as “found 200 positive reviews and 150 negative reviews” may not fully satisfy the information needs of different people. More useful information would be “This film is great for its novel originality” or “Poor acting, which makes the film awful”.

Opinion reason mining tries to identify one of the critical elements of online reviews to answer the question, “What are the reasons that the author of this review likes or dislikes the product?” To answer this question, we should extract not only sentences that contain opinion-bearing expressions, but also sentences with reasons why an author of a review writes the review (Cardie, Wiebe, Wilson, & Litman, 2003; Clarke & Terra, 2003; Li & Yamanishi, 2001; Stoyanov, Cardie, Litman, & Wiebe, 2004).

Kim and Hovy (2005) proposed a method for detecting opinion-bearing expressions. In their subsequent work (Kim & Hovy, 2006), they collected a large set of (review text, pros, cons) triplets from *epinions.com*, which explicitly state *pros* and *cons* phrases in their respective categories by each review’s author along with the review text. Their automatic labeling system first collects phrases in *pro* and *con* fields and then searches the main review text in order to collect sentences corresponding to those phrases. Then the system annotates this sentence with the appropriate “*pro*” or “*con*” label. All remaining sentences with neither label are marked as “*neither*”. After labeling all the data, they use it to train their *pro* and *con* sentence recognition system.

3.4. Other applications

Thomas, Pang, and Lee (2006) try to determine from the transcripts of US Congressional floor debates whether the speeches represent support of or opposition to proposed legislation. Mullen and Malouf (2006) describe a statistical sentiment analysis method on political discussion group postings to judge whether there is opposing political viewpoint to the original post. Moreover, there are some potential applications of sentiment detection, such as online message sentiment filtering, E-mail sentiment classification, web-log author’s attitude analysis, sentiment web search engine, etc.

4. Subjectivity classification

Subjectivity classification is a task to investigate whether a paragraph presents the opinion of its author or reports facts. In fact, most of the research showed there was very tight relation between subjectivity classification and document sentiment classification (Pang & Lee, 2004; Wiebe, 2000; Wiebe, Bruce, & O’Hara, 1999; Wiebe, Wilson, Bruce, Bell, & Martin, 2002; Yu & Hatzivassiloglou, 2003). Subjectivity classification can prevent the polarity classifier from considering irrelevant or even potentially misleading text. Pang and Lee (2004) find subjectivity detection can compress reviews into much shorter extracts that still retain polarity information at a level comparable to that of the full review.

Much of the research in automated opinion detection has been performed and proposed for discriminating between subjective and objective text at the document and sentence levels (Bruce & Wiebe, 1999; Finn, Kushmerick, & Smyth, 2002; Hatzivassiloglou & Wiebe, 2000; Wiebe, 2000; Wiebe et al., 1999; Wiebe et al., 2002; Yu & Hatzivassiloglou, 2003). In this section, we will discuss some approaches used to automatically assign one document as objective or subjective.

4.1. Similarity approach

Similarity approach to classifying sentences as opinions or facts explores the hypothesis that, within a given topic, opinion sen-

tences will be more similar to other opinion sentences than to factual sentences (Yu & Hatzivassiloglou, 2003). Similarity approach measures sentence similarity based on shared words, phrases, and WordNet synsets (Dagan, Shaul, & Markovitch, 1993; Dagan, Pereira, & Lee, 1994; Leacock & Chodorow, 1998; Miller & Charles, 1991; Resnik, 1995; Zhang, Xu, & Callan, 2002).

To measure the overall similarity of a sentence to the opinion or fact documents, we need to go through three steps. First, use IR method to acquire the documents that are on the same topic as the sentence in question. Second, calculate its similarity scores with each sentence in those documents and make an average value. Third, assign the sentence to the category (opinion or fact) for which the average value is the highest. Alternatively, for the frequency variant, we can use the similarity scores or count how many of them for each category, and then compare it with a predetermined threshold.

4.2. Naive Bayes classifier

Naive Bayes classifier is a commonly used supervised machine learning algorithm. This approach presupposes all sentences in opinion or factual articles as opinion or fact sentences.

Naive Bayes uses the sentences in opinion and fact documents as the examples of the two categories. The features include words, bigrams, and trigrams, as well as the part of speech in each sentence. In addition, the presence of semantically oriented (positive and negative) words in a sentence is an indicator that the sentence is subjective. Therefore, it can include the counts of positive and negative words in the sentence, as well as counts of the polarities of sequences of semantically oriented words (e.g., “++” for two consecutive positively oriented words). It also include the counts of parts of speech combined with polarity information (e.g., “JJ+” for positive adjectives), as well as features encoding the polarity (if any) of the head verb, the main subject, and their immediate modifiers.

Generally speaking, Naive Bayes assigns a document d_j (represented by a vector d_j^*) to the class c_i that maximizes $P(c_i|d_j^*)$ by applying Bayes’ rule as follow,

$$P(c_i|d_j^*) = \frac{P(c_i)P(d_j^*|c_i)}{P(d_j^*)} \quad (1)$$

where $P(d_j^*)$ is the probability that a randomly picked document d has vector d_j^* as its representation, and $P(c)$ is the probability that a randomly picked document belongs to class c .

To estimate the term $P(d_j^*|c)$, Naive Bayes decomposes it by assuming all the features in d_j^* (represented by $f_{i,i} = 1$ to m) are conditionally independent, i.e.,

$$P(c_i|d_j^*) = \frac{P(c_i)(\prod_{i=1}^m P(f_{i,i}|c_i))}{P(d_j^*)} \quad (2)$$

4.3. Multiple Naive Bayes classifier

The hypothesis of all sentences in opinion or factual articles as opinion or fact sentences is an approximation. To address this, multiple Naive Bayes classifier approach applies an algorithm using multiple classifiers, each relying on a different subset of features. The goal is to reduce the training set to the sentences that are most likely to be correctly labeled, thus boosting classification accuracy.

Given separate sets of features F_1, F_2, \dots, F_m , it train separate Naive Bayes classifiers C_1, C_2, \dots, C_m corresponding to each feature set. Assuming as ground truth the information provided by the document labels and that all sentences inherit the status of their document as opinions or facts, it first train C_1 on the entire training set,

then use the resulting classifier to predict labels for the training set. The sentences that receive a label different from the assumed truth are then removed, and train C_2 on the remaining sentences. This process is repeated iteratively until no more sentences can be removed. Yu and Hatzivassiloglou (2003) report results using five feature sets, starting from words alone and adding in bigrams, trigrams, part-of-speech, and polarity.

4.4. Cut-based classifier

Cut-based classifier approach put forward a hypothesis that, text spans (items) occurring near each other (within discourse boundaries) may share the same subjectivity status (Pang & Lee, 2004). Based on this hypothesis, Pang supplied his algorithm with pair-wise interaction information, e.g., to specify that two particular sentences should ideally receive the same subjectivity label. This algorithm uses an efficient and intuitive graph-based formulation relying on finding minimum cuts.

Suppose there are n items x_1, x_2, \dots, x_n to divide into two classes C_1 and C_2 , here access to two types of information:

$ind_j(x_i)$: Individual scores. It is the non-negative estimates of each x_i 's preference for being in C_j based on just the features of x_i alone;

$assoc(x_i, x_k)$: Association scores. It is the non-negative estimates of how important it is that x_i and x_k be in the same class.

Then, this problem changes to calculate the maximization of each item's score for one class: its individual score for the class it is assigned to, minus its individual score for the other class, then minus associated items into different classes for penalization. Thus, after some algebra, it arrives at the following optimization problem: assign the x_i to C_1 and C_2 so as to minimize the partition cost:

$$\sum_{x \in C_1} ind_2(x) + \sum_{x \in C_2} ind_1(x) + \sum_{x_i \in C_1, x_k \in C_2} assoc(x_i, x_k) \quad (3)$$

This situation can be represented in the following manner. Build an undirected graph G with vertices $\{v_1, \dots, v_n, s, t\}$; the last two are, respectively, the source and sink. Add n edges (s, v_i) , each with weight $ind_1(x_i)$, and n edges (v_i, t) , each with weight $ind_2(x_i)$. Finally, add (C_n^2) edges (v_i, v_k) , each with weight $assoc(x_i, x_k)$. A cut (S, T) of G is a partition of its nodes into sets $S = \{s\} \cup S'$ and $T = \{t\} \cup T'$, where $s \notin S', t \notin T'$. Its cost $cost(S, T)$ is the sum of the weights of all edges crossing from S to T . A minimum cut of G is one of *minimum cost*. Then, finding solution of this problem is changed into looking for a minimum cut of G .

5. Word sentiment classification

The task on document sentiment classification has usually involved the manual or semi-manual construction of semantic orientation word lexicons (Hatzivassiloglou & McKeown, 1997; Hatzivassiloglou & Wiebe, 2000; Lin, 1998; Pereira, Tishby, & Lee, 1993; Riloff, Wiebe, & Wilson, 2003; Turney & Littman, 2002; Wiebe, 2000), which built by word sentiment classification techniques. For instance, Das and Chen (2001) used a classifier on investor bulletin boards to see if apparently positive postings were correlated with stock price, in which several scoring methods were employed in conjunction with a manually crafted lexicon. Classifying the semantic orientation of individual words or phrases, such as whether it is positive or negative or has different intensities, generally using a pre-selected set of seed words, sometimes using linguistic heuristics (For example, Lin (1998) & Pereira et al. (1993) used linguistic co-locations to group words with similar uses or meanings).

Some studies showed that restricting features to those adjectives for word sentiment classification would improve perfor-

mance (Andreevskaia & Bergler, 2006; Turney & Littman, 2002; Wiebe, 2000). However, more researches showed most of the adjectives and adverb, a small group of nouns and verbs possess semantic orientation (Andreevskaia & Bergler, 2006; Esuli & Sebastiani, 2005; Gamon & Aue, 2005; Takamura, Inui, & Okumura, 2005; Turney & Littman, 2003).

Automatic methods of sentiment annotation at the word level can be grouped into two major categories: (1) corpus-based approaches and (2) dictionary-based approaches. The first group includes methods that rely on syntactic or co-occurrence patterns of words in large texts to determine their sentiment (e.g., Hatzivassiloglou & McKeown, 1997; Turney & Littman, 2002; Yu & Hatzivassiloglou, 2003 and others). The second group uses *WordNet* (<http://wordnet.princeton.edu/>) information, especially, synsets and hierarchies, to acquire sentiment-marked words (Hu & Liu, 2004a; Kim & Hovy, 2004) or to measure the similarity between candidate words and sentiment-bearing words such as *good* and *bad* (Kamps, Marx, Mokken, & de Rijke, 2004).

5.1. Analysis by conjunctions between adjectives

This method attempts to predict the orientation of subjective adjectives by analyzing pairs of adjectives (conjoined by *and*, *or*, *but*, *either-or*, or *neither-nor*) which are extracted from a large unlabelled document set. The underlying intuition is that the act of conjoining adjectives is subject to linguistic constraints on the orientation of the adjectives involved (e.g. *and* usually conjoins two adjectives of the same-orientation, while *but* conjoins two adjectives of opposite orientation). This is shown in the following three sentences (where the first two are perceived as correct and the third is perceived as incorrect) taken from Hatzivassiloglou and McKeown (1997):

“The tax proposal was simple and well received by the public”.

“The tax proposal was simplistic but well received by the public”.

“The tax proposal was simplistic and well received by the public”.

To infer the orientation of adjectives from analysis of conjunctions, a supervised learning algorithm can be performed as following steps:

1. All conjunctions of adjectives are extracted from a set of documents.
2. Train a log-linear regression classifier and then classify pairs of adjectives either as having the same or as having different orientation. The hypothesized same-orientation or different-orientation links between all pairs form a graph.
3. A clustering algorithm partitions the graph produced in step 2 into two clusters. By using the intuition that positive adjectives tend to be used more frequently than negative ones, the cluster containing the terms of higher average frequency in the document set is deemed to contain the positive terms.

The log-linear model offers an estimate of how good each prediction is, since it produces a value y between 0 and 1, in which 1 corresponds to same-orientation, and one minus the produced value y corresponds to dissimilarity. Same- and different-orientation links between adjectives form a graph. To partition the graph nodes into subsets of the same-orientation, the clustering algorithm calculates an objective function Φ scoring each possible partition P of the adjectives into two subgroups C_1 and C_2 as,

$$\Phi(P) = \sum_{i=1}^2 \left(\frac{1}{|C_i|} \sum_{x,y \in C_i, x \neq y} d(x,y) \right) \quad (4)$$

where $|C_i|$ is the cardinality of cluster i , and $d(x,y)$ is the dissimilarity between adjectives x and y .

In general, because the model was unsupervised, it required an immense word corpus to function.

5.2. Analysis by lexical relations

This method presents a strategy for inferring semantic orientation from semantic association between words and phrases. It follows a hypothesis that two words tend to be the same semantic orientation if they have strong semantic association. Therefore, it focused on the use of lexical relations defined in *WordNet* to calculate the distance between adjectives.

Generally speaking, we can define a graph on the adjectives contained in the intersection between a term set (For example, TL term set (Turney & Littman, 2003)) and *WordNet*, adding a link between two adjectives whenever *WordNet* indicates the presence of a synonymy relation between them, and defining a distance measure using elementary notions from graph theory. In more detail, this approach can be realized as following steps:

1. Construct relations at the level of words. The simplest approach here is just to collect all words in *WordNet*, and relate words that can be synonymous (i.e., they occurring in the same synset).
2. Define a distance measure $d(t_1, t_2)$ between terms t_1 and t_2 on this graph, which amounts to the length of the shortest path that connects t_1 and t_2 (with $d(t_1, t_2) = +\infty$ if t_1 and t_2 are not connected).
3. Calculate the orientation of a term by its relative distance (Kamps et al., 2004) from the two seed terms *good* and *bad*, i.e.,

$$SO(t) = \frac{d(t, bad) - d(t, good)}{d(good, bad)} \quad (5)$$

4. Get the result followed by this rules: The adjective t is deemed to belong to positive if $SO(t) > 0$, and the absolute value of $SO(t)$ determines, as usual, the strength of this orientation (the constant denominator $d(good, bad)$ is a normalization factor that constrains all values of SO to belong to the $[-1, 1]$ range).

5.3. Analysis by glosses

The characteristic of this method lies in the fact that it exploits the glosses (i.e. textual definitions) that one term has in an online “glossary”, or dictionary. Its basic assumption is that if a word is semantically oriented in one direction, then the words in its gloss tend to be oriented in the same direction (Esuli & Sebastiani, 2005; Esuli & Sebastiani, 2006a, 2006b). For instance, the glosses of *good* and *excellent* will both contain appreciative expressions; while the glosses of *bad* and *awful* will both contain derogative expressions.

Generally, this method can determine the orientation of a term based on the classification of its glosses. The process is composed of the following steps:

1. A seed set (S_p, S_n) , representative of the two categories *positive* and *negative*, is provided as input.
2. Search new terms to enrich S_p and S_n . Use lexical relations (e.g. synonymy) with the terms contained in S_p and S_n from a thesaurus, or online dictionary, to find these new terms, and then append them to S_p or S_n .
3. For each term t_i in $S'_p \cup S'_n$ or in the test set (i.e. the set of terms to be classified), a textual representation of t_i is generated by collating all the glosses of t_i as found in a machine-readable dictionary. Each such representation is converted into a vector by standard text indexing techniques.
4. A binary text classifier is trained on the terms in $S'_p \cup S'_n$ and then applied to the terms in the test set.

5.4. Analysis by both lexical relations and glosses

This method determines sentiment of words and phrases both relies on lexical relations (synonymy, antonymy and hyponymy) and glosses provided in *WordNet*.

Andreevskaia and Bergler (2006) proposed an algorithm named “STEP” (Semantic Tag Extraction Program). This algorithm starts with a small set of seed words of known sentiment value (*positive* or *negative*) and implements the following steps:

1. Extend the small set of seed words by adding synonyms, antonyms and hyponyms of the seed words supplied in *WordNet*. This step brings on average a 5-fold increase in the size of the original list with the accuracy of the resulting list comparable to manual annotations.
2. Go through all *WordNet* glosses, identifies the entries that contain in their definitions the sentiment-bearing words from the extended seed list, and adds these head words to the corresponding category – positive, negative or neutral.
3. Disambiguate the glosses with part-of-speech tagger, and eliminate errors of some words acquired in step 1 and from the seed list. At this step, it also filters out all those words that have been assigned contradicting.

In this algorithm, for each word we need compute a *Net Overlap Score* by subtracting the total number of runs assigning this word a negative sentiment from the total of the runs that consider it positive. In order to make the *Net Overlap Score* measure usable in sentiment tagging of texts and phrases, the absolute values of this score should be normalized and mapped onto a standard $[0, 1]$ interval. STEP accomplishes this normalization by using the value of the *Net Overlap Score* as a parameter in the standard fuzzy membership S-function (Zadeh, 1987). This function maps the absolute values of the *Net Overlap Score* onto the interval from 0 to 1, where 0 corresponds to the absence of membership in the category of sentiment (in this case, these will be the neutral words) and 1 reflects the highest degree of membership in this category. The function can be defined as follows,

$$S(u; \alpha; \beta; \gamma) = \begin{cases} 0 & \text{if } u \leq \alpha \\ 2\left(\frac{u-\alpha}{\gamma-\alpha}\right)^2 & \text{if } \alpha \leq u \leq \beta \\ 1 - 2\left(\frac{u-\alpha}{\gamma-\alpha}\right)^2 & \text{if } \beta \leq u \leq \gamma \\ 1 & \text{if } u \geq \gamma \end{cases} \quad (6)$$

where u is the *Net Overlap Score* for the word and α, β, γ are the three adjustable parameters: α is set to 1, γ is set to 15 and β , which represents a crossover point, is defined as $\beta = (\alpha + \gamma)/2 = 8$. Defined this way, the S-function assigns highest degree of membership (=1) to words that have the *Net Overlap Score* $u \geq 15$.

Net Overlap Score can be used as a measure of the words degree of membership in the fuzzy category of sentiment: the core adjectives, which had the highest *Net Overlap Score*, were identified most accurately both by STEP and by human annotators, while the words on the periphery of the category had the lowest scores and were associated with low rates of inter-annotator agreement.

5.5. Analysis by pointwise mutual information

The general strategy of this method is to infer semantic orientation from semantic association. The underlying assumption is that a phrase has a positive semantic orientation when it has good associations (e.g., “romantic ambience”) and a negative semantic orientation when it has bad associations (e.g., “horrific events”) (Turney, 2002).

The semantic orientation of a given word is calculated from the strength of its association with a set of positive words, minus the strength of its association with a set of negative words. More concretely, the strength of the semantic association between words can express by calculating their *pointwise mutual information* (PMI) value. So, it focuses on inferring the semantic orientation of a word from its statistical association with a set of positive and negative paradigm words. Given a term t , and seed term sets (S_p for positive set and S_n for negative set), the t 's orientation value $O(t)$ (where positive value means positive orientation, and higher absolute value means stronger orientation) is given by:

$$O(t) = \sum_{t_i \in S_p} PMI(t, t_i) - \sum_{t_i \in S_n} PMI(t, t_i) \quad (7)$$

Pointwise mutual information can be computed based on IR techniques. Term frequencies and co-occurrence frequencies are measured by querying a document set by means of a search engine with a “ t ” query, a “ t_i ” query, and a “ t NEAR t_i ” query, and using the number of matching documents returned by the search engine as estimates of the probabilities needed for the computation of PMI. In the *AltaVista* search engine (<http://www.altavista.com/>), the *NEAR* operator produces a match for a document when its operands appear in the document at a maximum distance of ten terms, in either order.

The paradigm words can be selected as following (Turney & Littman, 2003):

$S_p = \{good, nice, excellent, positive, fortunate, correct, superior\}$

$S_n = \{bad, nasty, poor, negative, unfortunate, wrong, inferior\}$

In addition, Gamon and Aue (2005) described an extension to the technique for the automatic identification and labeling of sentiment terms described in Turney and Littman (2003). Besides the basic assumption in Turney and Littman (2003), Turney (2002), Gamon and Aue (2005) adds a second assumption, namely that sentiment terms of opposite orientation tend not to co-occur at the sentence level. This additional assumption allows them to identify sentiment-bearing terms more reliably to some extent.

5.6. Analysis by general inquirer

General Inquirer (GI) is a system which lists terms as well as different senses for the terms. For each sense it provides a short definition as well as other information about the term. This includes tags that label the term as being *positive*, *negative*, *a negation term*, *an overstatement*, or *an understatement*. The labels are for each sense of a word.

For example, there are two senses of the word *fun* as seen in Table 1. One sense is a noun or adjective for *enjoyment* or *enjoyable*. The second sense is a verb that means *to ridicule or tease*, *to make fun of*. The first sense of the word is positive, while the second is negative. The entry also indicates that the first sense is more frequent than the second sense (estimated to occur 97% of the times while the second sense occurs only 3% of the times).

In addition, the *GI* dictionary includes *negations*, *intensifiers*, and *diminishers*. Table 2 shows the *GI* entries of the words *not*, *fantastic* and *barely* which are examples of a *negation*, an *overstatement* and an *understatement*, respectively.

The *GI* dictionary contains 1,915 positive senses and 2,291 negative senses. Kennedy and Inkpen (2006) add more positive and negative senses from *Choose the Right Word* (Hayakawa, 1994)

(hereafter *CTRW*). *CTRW* is a dictionary of synonyms, which lists nuances of lexical meaning. After adding them, they obtain 1955 positive senses and 2398 negative senses. There are 696 overstatements and 319 understatement in *GI*. When adding those from *CTRW*, they obtain 1269 overstatements and 412 understatement.

Kennedy and Inkpen (2006) used this approach to classify reviews based on the number of positive and negative terms they contain. They examined the effect of three types of valence shifters: *negations*, *intensifiers*, and *diminishers*. In: *GI intensifiers* are known as *overstatements* and *diminishers* are known as *understatements*. *Negations* are used to reverse the semantic polarity of a particular term, while *intensifiers* and *diminishers* are used to increase and decrease, respectively, the degree to which a term is positive or negative.

6. Document sentiment classification based on machine learning methods

6.1. Single domain

There are many possible approaches to identifying the actual polarity of a document. Here we talk about using supervised machine learning techniques to identify the likelihood of reviews having “*positive*” or “*negative*” polarity with respect to previously hand-classified training data. The key problem of this method includes two aspects: extracting feature and training classifier.

6.1.1. Extracting feature

Starting with a raw document (a portion of a web page in testing and training, and a complete web page for mining), then strip out HTML tags and separate the document into sentences. These sentences are optionally run through a parser before being split into single-word tokens. A variety of transformations can then be applied to this ordered list of lists. This is called feature extraction.

There are several methods for feature extraction:

1. Lexical filtering

Lexical filtering can apply to reviews, on the accuracy of statistical classifiers trained on such filtered data. There are mainly two different kinds of lexical filters used (Salvetti, Lewis, & Reichenbach, 2004): one based on hypernymy as provided by *WordNet* (Budanitsky & Hirst, 2001; Curran, 2002; Devitt & Vogel, 2004; Edmonds, 1999; Edmonds & Hirst, 2002; Fellbaum, 1998; Grefenstette, 1994; Justeson & Slava, 1993; Kamps & Maarten, 2002; Rapp, 2004; Riloff & Shepherd, 1997; Roark & Charniak, 1998; Taboada, 2006; Thelen & Riloff, 2002; Valitutti, Strapparava, & Stock, 2004), the other based on part-of-speech (*POS*) tags (Ait-Mokhtar, Chanod, & Roux, 2002; Brill, 1992; Brill, 1994; Brill, 1995; Losee, 2001; Ratnaparkhi, 1996; Schmid, 1994; Wiebe, Wilson, & Bell, 2001; Wiebe, Wilson, & Cardie, 2005; Wilks & Stevenson, 1998).

WordNet filter attempted to substitute synonyms in reviews by a set of likely synonyms and hypernymy generalization in *WordNet*, because it is uncommon to encounter repetitions of identical words in non-technical written text.

POS filter considered that individual phrases and words vary in their contribution to opinion polarity. It may even be said that only some part of the meaning of a word contributes to opinion polarity. Any portion that does not contribute to sentiment detection is noise. *POS* filters were developed to reduce these noises.

Table 1

GI entries for the word fun.

Fun (sense 1)	H4Lvd Positiv Pstv Pleasur Exprsv WlBPsyc WlBTot Noun PFREQ 97% noun-adj: Enjoyment, enjoyable
Fun (sense 2)	H4Lvd Negativ Ngvtv Hostile ComForm SV RspLoss RspTot SUPV 3% idiom-verb: Make fun (of) – to tease, parody

Table 2

GI entries for the words not, fantastic and barely.

Not	H4Lvd negate NotLw LY – adv: expresses negation NotLw LY adv: expresses negation
Fantastic	H4Lvd Positiv Pstv Virtue Ovrst EVAL PosAff Modif – Virtue Ovrst EVAL PosAff Modif
Barely	H4Lvd Undrst Quan If LY – Quan If LY

2. Appraising adjective

Appraising adjective method (Whitelaw, Argamon, & Garg, 2005; Whitelaw, Garg, & Argamon, 2005) focuses on the extraction and analysis of adjectival appraisal groups headed by an appraising adjective (such as “beautiful” or “boring”) and optionally modified by a sequence of modifiers (such as “very”, “sort of”, or “not”). It made a more detailed semantic analysis of attitude expressions, in the form of a well-designed taxonomy of attitude types and other semantic properties. Furthermore, it treats “atomic units” of such expressions not with individual words, but with appraisal groups: coherent groups of words that express together a particular attitude, such as “extremely boring”, or “not really very good”. This method can be described as following steps:

1. Build a lexicon using semi-automatic techniques, gathering and classifying adjectives and modifiers to categories in several taxonomies of appraisal attributes.
2. Extract adjectival appraisal groups from texts and compute their attribute values according to this lexicon.
3. Represent documents as vectors of relative frequency features using these groups.
4. Train a support vector machine algorithm discriminating positively from negatively oriented test documents.

Beineke, Hastie, and Vaithyanathan (2004) extend this procedure by extract a pair of derived features that are linearly combined to predict sentiment. This perspective allows to improving upon previous methods, primarily through two strategies: incorporating additional derived features into the model and, where possible, using labeled data to estimate their relative influence. Matsumoto, Takamura, and Okumura (2005) used text-mining techniques to extract frequent word sub-sequences and dependency sub-trees from sentences in a document dataset and used them as features of support vector machines.

6.1.2. Training classifier

Several classical text classifiers, such as K-Nearest Neighbor, Winnow, Naïve Bayes, Maximum Entropy and Support Vector Machine (SVM), are used in machine learning method for document sentiment classification.

Pang et al. (2002) applied Naïve Bayes, Maximum Entropy and Support Vector Machine classification techniques to the identification of the polarity of movie reviews. Their best result (82.9% accuracy) was obtained by using unigrams and Support Vector Machine.

In fact, most of the literatures showed that SVM and Naïve Bayes are perfect methods in single domain document sentiment classification (Aue and Gamon, 2005; Beineke et al., 2004; Kennedy and Inkpen, 2006; Lin et al., 2006; Matsumoto et al., 2005; Mullen and Collier, 2004; Pang and Lee, 2004; Pang et al., 2002; Read et al., 2005; Salvetti et al., 2004; Whitelaw, Garg, et al., 2005).

6.2. Multiple domains

Document sentiment classification is a very domain-specific problem: classifiers trained in one domain do not perform well in others Charlotta (2004), Nigam, McCallum, and Thrun (2000),

Lafferty, McCallum, and Pereira (2001), Avrim and Chawla (2001). The main factor is that the polarity of sentiment words may change with the domain too. For instance, the word “small” in house review is negative (e.g. “The bedroom is very small”), while in cell-phone review is positive (e.g. “The Nokia N3100 is so small as to be put in any pockets”).

Tan, Wu, Tang, and Cheng (2007) attempted to tackle domain-transfer problem by combining old-domain labeled examples with new domain unlabeled ones. Their basic idea is to use old-domain-trained classifier (“old classifier” for brevity) to label top n most informative unlabeled examples in new domain and learn a new classifier based on these selected examples (n is a pre-defined number indicating how many examples in new domain shall be picked out as informative ones). Detailed algorithm for proposed scheme is:

1. Train a base classifier using labeled data in old-domain.
2. Label some informative unlabeled ones in new domain.
3. Train a new classifier based on these selected examples.
4. Classify examples in new domain using new classifier.

The experimental results demonstrate their proposed scheme can boost the accuracy of the base sentiment classifier on new domain.

Aue and Gamon (2005) surveyed four different approaches to customizing a sentiment classification system to a new target domain with a small amount of labeled data. Read et al. (2005) proposed a novel source of training data based on the language used in conjunction with emoticons in *Usenet* newsgroups. Then they used emoticons-labeled data to train a classifier, which has the potential to reduce dependent of domain, topic and time.

7. Opinion extraction

Analysis of favorable and unfavorable opinions is a task requiring high intelligence and deep understanding of the textual context, drawing on common sense and domain knowledge as well as linguistic knowledge. The interpretation of opinions can be debatable even for humans. For example, when we tried to determine if each specific document was on balance favorable or unfavorable toward a subject after reading an entire group of such documents, we often found it difficult to reach a consensus, even for very small groups of evaluators.

Many researchers think it is too coarse to compute a unit of an opinion for a whole document (Bai, Padman, & Airoidi, 2004; Berhard, Yu, Thornton, Hatzivassiloglou, & DanJurafsky, 2004; Bunesco & Mooney, 2004; Daille, 1996; Didier, 1995; Etzioni et al., 2004; Freitag & McCallum, 2000; Jacquemin & Bourigault, 2001; Luca & Mazzini, 2002; Riloff & Jones, 1999; Wilson, Wiebe, & Hwa, 2004; Zhai & Liu, 2005). Turney (2002) makes a similar point, noting that for reviews, “the whole is not necessarily the sum of the parts”. Even in a single sentence, a holder might express two different opinions. It seems necessary using more sophisticated techniques to determine the focus of each sentence, so that one can decide whether the author is talking about the topic (Dave, Lawrence, & Pennock, 2003; Fei, Liu, & Wu, 2004; Kim & Hovy, 2004; Ku, Wu, Lee, & Chen, 2005; Mei, Ling, Wondra, Su, & Zhai, 2007).

Consequently, opinion extraction plays a very important role in sentiment detection. It not only focuses on extracting opinion information from reviews, but also on extracting relation between opinion and document topic.

7.1. Opinion information extraction

The opinion information we mainly discussed in this paper includes opinion-bearing word and opinion holder. An opinion-bearing word is a word or a phrase that carries a positive or negative sentiment directly such as “good”, “bad”, “foolish”, “virtuous”, etc. An opinion holder is an entity (person, organization, country, or special group of people) who expresses explicitly or implicitly the opinion contained in the sentence. For instance:

“According to the review in Internet, the keypad of Ericsson W810i is easy to use”.

In above sentence, “the review in Internet” is an opinion holder; “easy” is an opinion-bearing word.

7.1.1. Opinion-bearing word extraction

Opinions can be recognized from various granularities such as a word, a sentence, a text, or even multiple texts, and each is important. Here we focus on word level opinion detection, i.e., finding words or phrases that carry a positive or negative sentiment (opinion-bearing word) from subjectivity sentences or paragraphs. Actually, opinion-bearing word is the smallest unit of opinion that can thereafter be used as a clue for sentence-level or text-level opinion detection.

How to extract opinion-bearing words from document? A straightforward way can be conducted by following steps:

1. Collect sentiment words from sentiment lexicon as opinion-bearing seed words. Opinion-bearing seed words can be collected from several sources: General Inquirer (GI), Dictionary of Affect of Language (DAL), and WordNet (Kim & Hovy, 2006; Yi, Nasukawa, Bunescu, & Niblack, 2003).
2. Expand those selected opinion-bearing seed words of each sentiment class by collecting synonyms from WordNet. However, it cannot simply assume that all the synonyms of positive words are positive since most words could have synonym relationships with both positive and negative classes. It should calculate the closeness of a given word to each category and determine the most probable class (Kim & Hovy, 2006).
3. Refine some of the sentiment patterns from sentiment lexicon and training datasets (Yi et al., 2003). For GI and DAL, the sentiment verb extraction is the same as the opinion-bearing seed words extraction. For WordNet, the sentiment verb can be extracted from the *emotion cluster*. The other sentiment patterns can be manually refined from the training datasets. Sentiment pattern database contains sentiment extraction patterns for sentence predicates.
4. Extract sentences and text fragments from input documents containing subjectivity information (the approach we have discussed in Section 4). Apply sentiment analysis with expanded opinion-bearing seed words and sentiment patterns to those subjectivity sentences and text fragments. At last, the opinion-bearing words were extracted.

7.1.2. Opinion holder extraction

The goal of opinion holder extraction is to identify direct and indirect sources of opinions, emotions, sentiments, and other private states that are expressed in text. Identifying opinion sources (or opinion holder) will be especially critical for opinion-oriented question–answering systems (e.g., systems that answer questions of the form “Who feels about . . . ?”) and opinion-oriented summa-

rization systems, both of which need to distinguish the opinions of one source from another.

There are mainly three research points in opinion holder extraction. The first is: given a sentence, identifying opinion sources in it (Choi, Cardie, Riloff, & Patwardhan, 2005); the second is: given an opinion expression in a sentence, identifying its corresponding opinion sources (Kim & Hovy, 2006); the third is: given an opinion expression and a source, determines whether the source and the opinion expression is correspond (Choi, Breck, & Cardie, 2006).

1. The first research point

The first research point learns patterns of opinion sources using a graphical model and extraction pattern learning. It views the opinion holder extraction problem as an information extraction task and adopts a hybrid approach that combines CRFs (Conditional Random Fields) and a variation of *AutoSlog* (a supervised extraction pattern learner that takes a training corpus of texts and their associated answer keys as input) (Riloff, 1996). While CRFs identifies source and *AutoSlog* learns extraction patterns. It starts with a sequence of words (x_1, x_2, \dots, x_n) in a sentence and then:

- (1) Generate a sequence of labels (y_1, y_2, \dots, y_n) indicating whether the word is a holder or not.
- (2) Presents a new variation of *AutoSlog*, *AutoSlog-SE*, which generates patterns to extract sources.

This algorithm is not perfect, however, so the resulting set of patterns needs to be manually reviewed by a person. In order to build a fully automatic system which has no use for manual review, Choi et al. (2005) combined *AutoSlog*'s heuristics with statistics from the annotated training data to create a fully automatic supervised learner.

2. The second research pointThe second research point uses classification and ranking to model the problem with Maximum Entropy (ME) (Kim & Hovy, 2006). Classification allocates each holder candidate to a set of pre-defined classes while ranking selects a single candidate as answer. It chooses the most probable candidate via a conditional probability. This method acts as following steps.

- (1) Generate all possible holder candidates, given a sentence and an opinion expression $\langle E \rangle$. After parsing the sentence, it extracts features such as the syntactic path information between each candidate $\langle H \rangle$ and the expression $\langle E \rangle$ and a distance between $\langle H \rangle$ and $\langle E \rangle$.
- (2) Rank holder candidates according to the score obtained by the ME ranking model, and pick the candidate with the highest score.

Given a set of holder candidates $\{h_1, h_2, \dots, h_n\}$ and opinion expression e . The conditional probability $P(h|\{h_1, h_2, \dots, h_n\}, e)$ can be calculated based on K feature functions $f_k(h|\{h_1, h_2, \dots, h_n\}, e)$, as follows,

$$h = \operatorname{argmax}_h [P(h|\{h_1, h_2, \dots, h_n\}, e)] \\ = \operatorname{argmax}_h \left[\sum_{k=1}^K \lambda_k f_k(h|\{h_1, h_2, \dots, h_n\}, e) \right] \quad (8)$$

where each λ_k is a model parameter indicating the weight of its feature function.

3. The third research point

The third research point (Choi et al., 2006) aimed to identify the opinion holder via extracting relations between opinion expression entities and source entities. That is, given opinion expression O_i and source S_j , it determines whether S_j is the source of opinion expression O_i . The global inference procedure is implemented via integer linear programming (ILP) to produce an optimal and coherent extraction of entities and relations.

ILP formulation group consists of an objective function and a set of equality and inequality constraints among variables. The following formulation is the objective function of binary ILP,

$$f = \sum_i (w_{o_i} O_i) + \sum_i (w_{o_i}^* O_i^*) + \sum_j (w_{s_j} S_j) + \sum_j (w_{s_j}^* S_j^*) + \sum_{ij} (w_{l_{ij}} L_{ij}) + \sum_{ij} (w_{l_{ij}}^* L_{ij}^*) \quad (9)$$

$$\forall i, O_i + O_i^* = 1; \quad \forall j, S_j + S_j^* = 1, \quad \forall i, j, L_{ij} + L_{ij}^* = 1 \quad (10)$$

where f is the objective function of *ILP*. O_i and O_i^* are two variables for each opinion entity, $O_i = 1$ means to extract the opinion entity, and $O_i^* = 1$ means to discard the opinion entity. w_{o_i} and $w_{o_i}^*$ are weights for O_i and O_i^* , respectively, which are computed based on the labels of the adjacent variables of the CRFs (Choi et al., 2006). Likewise, S_j and S_j^* are two variables for each source entity to indicate whether it had been extracted, their weights w_{s_j} and $w_{s_j}^*$ are computed in the same way as opinion entities; L_{ij} and L_{ij}^* are two variables for each link relation between O_i and S_j . $L_{ij} = 1$ indicate O_i and S_j both be extracted, otherwise, $L_{ij}^* = 1$. Their weights $w_{l_{ij}}$ and $w_{l_{ij}}^*$ are based on probabilities from the binary link classifier. The following is a set of equality and inequality constraints among variables.

$$\forall i, O_i = \sum_j L_{ij} \quad (11)$$

$$\forall j, S_j + A_j = \sum_i L_{ij} \quad (12)$$

$$\forall j, A_j - S_j \leq 0 \quad (13)$$

$$\forall i, j, i < j, X_i + X_j = 1, \quad X \in \{S, O\} \quad (14)$$

Formula (11) enforces that only one link can emanate from an opinion entity. Formulas (12) and (13) together allow a source to link to at most two opinions, where A_j is an auxiliary variable between 0 and 1. A_j can be assigned to 1 only if S_j is already assigned to 1. Formula (14) is used to restrict all pairs of entities with overlapping spans.

7.2. Opinion-topic relation extraction

Opinion-topic relation refers to relationship between opinion expression (opinion-bearing words) and document topic (or feature of topic).

A feature of a topic is a term that satisfies one of the following relationships:

- A part-of relationship with the given topic.
- An attribute-of relationship with the given topic.
- An attribute-of relationship with a known feature of the given topic.

For instance, “I found the Sony Ericsson W810i is a good mobile phone. The keypad is easy to use and texting is very simple as the buttons are small but well defined. The screen is bright and clear with good resolution. The software on this phone is excellent. I have had absolutely no problems with it what so ever, it never crashes, freezes or otherwise upsets me.

The battery life on this phone is excellent. I have had mine for 18 months, with it rarely being switched off in all this time. In my normal use (only about 30–60 min calls a day) it will last for about 3–4 days before needing a charge. Charging is very quick too taking only around an hour to fully charge from about 10%.

All in all an excellent little phone with very few faults. Recommended!”

In above text, “good”, “easy to use”, “bright” etc., are *opinion expression*, “Sony Ericsson W810i” is *topic*, and “keypad”, “screen”, “software” etc., are *features of topic*.

7.2.1. Feature term extraction

Yi et al. (2003) put forward the following three candidates as feature term to be extracted:

1. Base Noun Phrases (BNP). BNP restricts the candidate feature terms to one of the following base noun phrase (BNP) patterns: *NN*, *NN NN*, *JJ NN*, *NN NN NN*, *JJ NN NN*, *JJ JJ NN*, where *NN* and *JJ* are the part-of-speech (POS) tags for nouns and adjectives, respectively.
2. Definite Base Noun Phrases (dBNP). dBNP further restricts candidate feature terms to definite base noun phrases, which are noun phrases of the form defined above that are preceded by the definite article “the”. Given that a document is focused on a certain topic, the definite noun phrases referring to topic features do not need any additional constructs such as attached prepositional phrases or relative clauses, in order for the reader to establish their referent. Thus, the phrase “the battery,” instead of “the battery of the digital camera,” is sufficient to infer its referent.
3. Beginning Definite Base Noun Phrases (bBNP). bBNP refers to dBNP at the beginning of sentences followed by a verb phrase. This heuristic is based on the observation that, when the focus shifts from one feature to another, the new feature is often expressed using a definite noun phrase at the beginning of the next sentence.

They developed and tested two feature term selection algorithms based on a mixture language model and likelihood ratio, while the Likelihood Test method gets better result. Following is principle of the Likelihood Test method: Let D_+ be a collection of documents focused on a topic T , D_- those not focused on T , and *bnp* a candidate feature term extracted from D_+ . Then, the likelihood ratio $-2 \log \lambda$ is defined as follows,

$$-2 \log \lambda = -2 \log \frac{\max_{p_1 \leq p_2} L(p_1, p_2)}{\max_{p_1, p_2} L(p_1, p_2)} \quad (15)$$

$$p_1 = p(d \in D_+ | \overline{bnp} \in d)$$

$$p_2 = p(d \in D_+ | \overline{bnp} \in d)$$

where $L(p_1, p_2)$ is the likelihood of seeing *bnp* in both D_+ and D_- . The higher the value of $-2 \log \lambda$, the more likely the *bnp* is relevant to the topic T . For each *bnp*, compute the likelihood score, $-2 \log \lambda$, as defined in formula (15). Then, sort *bnp* in decreasing order by their likelihood score. Feature terms are all *bnp*'s whose likelihood ratio satisfying a pre-defined confidence level. Alternatively simply only the top N *bnp*'s can be selected.

7.2.2. Makes (topic | feature term, opinion) association

In order to achieve high precision, we need focus on identifying semantic relationships between opinion expressions and topic (or feature of topic) terms, i.e. extracting opinion-bearing words associated with polarity of positive or negative for specific topic (or feature of topic) from a document (Agrawal & Srikant, 1994; Jon & Tardos, 2002; Liu, Hsu, & Ma, 1998; Rosario & Hearst, 2004).

In order to identify opinion expressions and analyze their semantic relationships with the topic (or feature of topic) term, following natural language processing techniques play an important role (Nasukawa & Yi, 2003):

1. POS tagging
POS tagging can disambiguate some polysemous expressions such as “like,” which denote sentiment only when used as a verb instead of as an adjective or preposition.
2. Syntactic parsing
Syntactic parsing is used to identify relationships between sentiment expressions and the subject term. Furthermore, in order to

maintain robustness for noisy texts from various sources such as the WWW, it need to use a shallow parsing framework that identifies phrase boundaries and their local dependencies in addition to POS tagging, instead of using a full parser that tries to identify the complete dependency structure among all of the terms.

Yi et al. (2003) extracts *ternary expressions* (*T-expressions*) and *binary expressions* (*B-expressions*), in order to make (topic | feature term, opinion) association. There are two types of *T-expressions*:

1. positive or negative sentiment verbs: (<target, verb, "")
2. trans verbs: (target, verb, source) And one format of *B-expressions*: (adjective, target)

For each opinion expressions detected, its target and final polarity can be determined by sentiment pattern database (Sentiment pattern database contains sentiment extraction patterns for sentence predicates.) If no corresponding sentiment pattern is available, the *B-expressions* can be created for making the sentiment assignment.

From a *T-expression*, sentiment of the verb (for sentiment verbs) or source (for trans verb), and from a *B-expression*, sentiment of the adjective, is assigned to the target.

8. Evaluation of sentiment detection

As for sentiment analysis systems, the evaluation of sentiment classifiers is typically conducted experimentally, rather than analytically. The reason is that, in order to evaluate a system analytically (e.g., proving that the system is correct and complete), we would need a formal specification of the problem that the system is trying to solve (e.g., with respect to what correctness and completeness are defined), and the central notion of sentiment detection is. The experimental evaluation of a classifier usually measures its effectiveness (rather than its efficiency), that is, its ability to take the right classification decisions.

8.1. Sentiment classification effectiveness

8.1.1. Precision and recall

Precision is the ratio of correct cases within the system outputs. Recall is the ratio of correct cases that the system assigned compared to the base of all cases where a human analyst associated either positive or negative sentiments manually. In other words, *precision* and *recall* are calculated with the following formulas:

A = number of all cases that the system assigned either a positive or negative sentiment

B = number of all cases that the human assigned either a positive or negative sentiment

C = number of correct cases in the system output based on the manual judgment

$$\text{Precision} = \frac{C}{A} \quad \text{Recall} = \frac{C}{B} \quad (16)$$

The traditional *F-measure* value can compute as follow formula:

$$F = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{(\text{Precision} + \text{Recall})} \quad (17)$$

This is also known as the *F1 measure*, because *recall* and *precision* are evenly weighted.

Leshed and Kaye (2006) use precision and recall to estimate Blogger mood recognition. Ye, Shi, and Li (2006) distinguish the precision and recall ratios of positive and negative reviews.

Classification effectiveness is usually measured in terms of the classic IR notions, like correlation coefficient, relative error, precision, recall, accuracy, etc. However, these values did not

seem to provide any better differentiation than simple accuracy. In the domain of sentiment classification of reviews it is often acceptable to sacrifice recall for accuracy. This result is particularly interesting for applications that rely on web data, because the customer is not always interested in having all the possible reviews, but many times is interested in having just a few positive and a few negative. From this perspective accuracy is more important than recall.

8.1.2. Correlation coefficient and relative error

The correlation coefficient indicates how accurate the sentiment classification is, showing to what degree the fluctuation patterns (e.g., detection of peaks and drops) of a sentiment are predicted by the model (Mishne & de Rijke, 2006). A correlation coefficient of 1 means that there is a perfect linear relation between the prediction and the actual values; whereas a correlation coefficient of 0 means that the prediction is completely unrelated to the actual values. The correlation coefficient is a standard measure of the degree to which two variables are linearly related, and is defined as

$$\text{CorrCoefficient} = \frac{S_{PA}}{S_p \cdot S_A} \quad (18)$$

where

$$S_{PA} = \frac{\sum_i (p_i - \bar{p}) \cdot (a_i - \bar{a})}{n - 1} \quad (19)$$

$$S_p = \frac{\sum_i (p_i - \bar{p})^2}{n - 1}, \quad S_A = \frac{\sum_i (a_i - \bar{a})^2}{n - 1}$$

where p_i is the estimated value for instance i , a_i is the actual value for instance i , \bar{x} is the average of x , and n is the total number of instances.

The relative error denotes the mean difference between the actual values and the estimated ones, and is defined as:

$$\text{RelError} = \frac{\sum_i (|p_i - a_i|)}{\sum_i (|a_i - \bar{a}|)} \quad (20)$$

8.2. Benchmarks for sentiment detection

Generally speaking, different experiments used for comparing only if the experiments satisfy following conditions:

1. On exactly the same collection (i.e., same documents and same categories);
2. With the same “split” between training set and test set;
3. With the same evaluation measure, and the same parameter values (if have).

That is to say, a lack of these three conditions may make the experimental results hardly comparable between each other (see Table 3).

We cannot cover the experiments performed on one collection, because it is hardly to find enough number of authors used the same collection in same experimental conditions. Tables 4–6 list the results of all experiments known to us performed on four major datasets, which HM and GI datasets are used for word sentiment classification, while IMDB and polarity datasets are used for document sentiment classification.

8.2.1. Benchmarks for word sentiment classification

The HM dataset consists of 1336 adjectives, 657 positive and 679 negative (Hatzivassiloglou & McKeown, 1997).

The GI dataset is a list of labeled words extracted from the General Inquirer lexicon (Stone, Dunphy, Smith, & Ogilvie, 1966).

Table 3

Comparative results among different literatures obtained on HM dataset (boldface indicates the best performer on the collection).

Author & literature	Training set	Accuracy
Hatzivassiloglou and McKeown (1997)	A corpus of 21 million words	.924
Turney and Littman (2003)	SO-PMI with three corpus (AV-ENG, AV-CA, TASA)	.982
	SO-LSA with corpus TASA	.889
Esuli and Sebastiani (2005)	WordNet	.874

Table 4

Comparative results among different literatures obtained on GI dataset (boldface indicates the best performer on the collection).

Author & literature	Training set	Accuracy
Turney and Littman (2002)	SO-PMI-IR with a two-billion word corpus	.894
	SO-LSA with a 10-million word corpus	.817
Kamps et al. (2004)	WordNet	.787
Turney and Littman (2003)	SO-PMI with three corpus (AV-ENG, AV-CA, TASA)	.971
	SO-LSA with corpus TASA	.820
Esuli and Sebastiani (2005)	WordNet	.881
Takamura et al. (2005)	WordNet	.915

Table 5

Comparative results among different literatures obtained on IMDb dataset (boldface indicates the best performer on the collection).

Author & literature	Classifier used	Accuracy
Pang et al. (2002)	NB	.815
	ME	.810
	SVM	.829
Salveti et al. (2004)	NB	.795
	Markov model	.805
Mullen and Collier (2004)	SVM	.860
Beineke et al. (2004)	NB	.659
Matsumoto et al. (2005)	SVM	.883

Table 6

Comparative results among different literatures obtained on polarity dataset (boldface indicates the best performer on the collection).

Author & literature	Classifier used	Accuracy
Pang and Lee (2004)	SVM	.872
Whitelaw, Garg et al., (2005)	SVM	.902
Matsumoto et al. (2005)	SVM	.937
Aue and Gamon (2005)	SVM	.905
Read et al. (2005)	NB	.789
	SVM	.815
Kennedy and Inkpen (2006)	SVM	.862

It includes 3596 adjectives, adverbs, nouns, and verbs, in which 1614 are positive and 1982 are negative.

8.2.2. Benchmarks for document sentiment classification

The Internet Movie Database (IMDb, <http://reviews.imdb.com/Reviews/>) dataset consists of 27,000 movie reviews in HTML form, using 35 different rating scales such as A..F or 1..10 in addition to the common 5 star system.

The Polarity dataset contains 1000 positive and 1000 negative reviews all written before 2002, with a cap of 20 reviews per author (312 author total) per category.

9. Conclusion

Sentiment detection has a wide variety of applications in information systems, including classifying reviews, distinguishing synonyms and antonyms, extending the capabilities of search engines, summarizing reviews, tracking opinions in online discussions, and analyzing survey responses. There are likely to be many other applications that we have not anticipated. This paper discusses four related problems, i.e., subjectivity classification, word senti-

ment classification, document sentiment classification based on machine learning techniques, and opinion extraction problem.

Although we were able to obtain fairly good results for the review classification task through the choice of appropriate features and metrics, but we identified a number of issues that make this problem difficult.

1. Since sentiments can be expressed with various expressions including indirect expressions that require common sense reasoning to be recognized as a sentiment, it's been a challenge to analyze the complex structures of sentences in the input context that negates the local sentiment for the whole.
2. Some reviewers use terms that have negative connotations, but then write an equivocating final sentence explaining that overall they were satisfied. Mixed reviews introduce considerable noise to the problem of scoring words.
3. Although the overall opinion about a topic is useful, it is only a part of the information of interest. Document level sentiment classification fails to detect sentiment about individual aspects of the topic. In reality, for example, though one could be generally happy about his car, he might be dissatisfied by the engine noise. To the manufacturers, these individual weaknesses and strengths are equally important to know, or even more valuable than the overall satisfaction level of customers.
4. The association of the extracted sentiment to a specific topic is difficult. Most statistical opinion extraction algorithms perform poorly in this aspect. They either assume the topic of the document is known at the fore, or simply associate the opinion to a topic term co-existing in the same context. But in fact, a document (or even a portion of a document as small as a sentence) may discuss multiple topics and contain sentiment about multiple topics.
5. An accurate identification of semantic orientation requires analysis of units larger than individual words; it requires understanding of the context in which those words appear. To this end, we intend to use Rhetorical Structure Theory to impose on the text a structure that indicates the relationships among its rhetorical units.

In future, more work is needed on further improving and refining techniques mentioned in this paper, and to deal with the outstanding problems identified above. The success story of sentiment detection is also going to encourage an extension of its methods and techniques to neighboring fields of application. A variety of steps can be taken to extend this work:

1. In addition to using a larger amount of discriminating terms, additional non-content attributes can be used for sentiment detection, e.g., collect more text marked-up with emoticons. Features which seem promising are the use of emoticons – textual representations of facial expressions – as well as sentiment values of the individual words, and other features.
2. Learn ways to separate out review topic and subtopics (or attributes) within reviews and treat them differently.
3. Explore the feasibility of integrating a full parser and various discourse processing methods to better sentence structure analysis, thus better relationship analysis, i.e., associating topic with holders and topic with feature terms.
4. Combine manual and automated generation of lexicon techniques, build a more general sentiment lexicon, modify and add sentiment terms for further domains with domain information and sentiment intensity (or sentiment strength), which will be used in multi-domain and multi-class sentiment classification.
5. Look into monitoring (or tracing) of opinions, take the time sequence of reviews into account, and draw the outline of sentiment mutative trend. An opinion tracking system provides not only text-based and graph-based opinion summaries, but also the trend of opinions from many information sources.
6. Sentiment clustering. It is possible to cluster sentiment according to the correlation between the temporal behavior of certain sentiment and the temporal behavior of opinion-bearing word frequencies in the text (measuring the occurrences of words over time).
7. Opinion question answering. For instance, who is the opinion holder? Why did the holder propose this opinion? How did the holder express this opinion, and what was the final attitude?

Acknowledgements

This work was mainly supported by special fund of Chinese Academy of Sciences, “Research on Opinion Mining of Web Text”, under Grant Number 0704021000 and another four Projects, i.e., 2006AA010105, 2007AA01Z416, 2007CB311100 and 2007AA01Z441.

References

- Agrawal, R., & Srikant, R. (1994). Fast algorithm for mining association rules. In *VLDB'94*.
- Ait-Mokhtar, S., Chanod, J. P., & Roux, C. (2002). Robustness beyond shallowness: Incremental deep parsing. *Natural Language Engineering*, 8(2–3), 121–144.
- Andreevskaia, A., & Bergler, S. (2006). Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In *Proceedings EACL-06, Trento, Italy*.
- Argamon, S., Koppel, M., & Avneri, G. (1998). Routing documents according to style. In *First international workshop on innovative information systems*.
- Aue, A., & Gamon, M. (2005). Customizing sentiment classifiers to new domains: A case study. In *Proceedings of RANLP*.
- Avrim, B., & Chawla, S. (2001). Learning from labeled and unlabeled data using graph mincuts. In *Proceedings 18th international conference on machine learning (ICML)* (pp. 19–26).
- Bai, X., Padman, R., & Airoidi, E. (2004). Sentiment extraction from unstructured text using tabu search-enhanced Markov blanket. In *Proceedings of the international workshop on mining for and from the semantic web* (pp. 24–35).
- Beineke, P., Hastie, T., Vaithyanathan, & S. (2004). The sentimental factor: Improving review classification via human-provided information. In *Proceedings of the 42nd ACL conference*.
- Bethard, S., Yu, H., Thornton, A., Hatzivassiloglou, V., & DanJurafsky (2004). Automatic extraction of opinion propositions and their holders. In *AAAI spring symposium on exploring attitude and affect in text*.
- Brill, E. (1992). A simple rule-based part of speech tagger. In *Proceedings of ACL conference on applied natural language processing, Trento, Italy*.
- Brill, E. (1994). Some advances in transformation-based parts of speech tagging. In *AAAI*.
- Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4), 543–565.
- Bruce, R. F., & Wiebe, J. (1999). Recognizing subjectivity: A case study in manual tagging. *Natural Language Engineering*, 5(02), 187–205.
- Budanitsky, A., & Hirst, G. (2001). Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and other lexical resources, second meeting of the NAACL, Pittsburgh*.
- Bunescu, R. C., & Mooney, R. J. (2004). Collective information extraction with relational markov networks. In *ACL'2004*.
- Cardie, C., & Wiebe, J., Wilson, T., & Litman, D. (2003). Combining low-level and summary representations of opinions for multi-perspective question answering. In *AAAI spring symposium on new directions in question answering* (pp. 20–27).
- Chaovalit, P., & Zhou, L. (2005). Movie review mining: A comparison between supervised and unsupervised classification approaches. In *Proceedings of the 38th Hawaii international conference on system sciences* (pp.1–9). Big Island, Hawaii: IEEE.
- Charlotta, E. (2004). *Topic dependence in sentiment classification*. Master's thesis, University of Cambridge.
- Choi, Y., Cardie, C., Riloff, E., & Patwardhan, S. (2005). Identifying sources of opinions with conditional random fields and extraction patterns. In *Proceedings of HLT/EMNLP*.
- Choi, Y., Breck, E., & Cardie, C. (2006). Joint extraction of entities and relations for opinion recognition. In *Proceedings of the 2006 conference on empirical methods in natural language processing (EMNLP 2006), Sydney* (pp. 431–439).
- Clarke, C. L. A., & Terra, E. L. (2003). Passage retrieval vs. document retrieval for factoid question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval, Toronto, Canada* (pp. 427–428).
- Curran, J. R. (2002). Ensemble methods for automatic thesaurus extraction. In *Proceedings of the 2002 conference on empirical methods in natural language processing, Philadelphia, PA, USA* (pp. 222–229).
- Dagan, I., Shaul, M., & Markovitch, S. (1993). Contextual word similarity and estimation from sparse data. In *Proceedings of ACL-93, Columbus, Ohio* (pp. 164–171).
- Dagan, I., Pereira, F., & Lee, L. (1994). Similarity-based estimation of word co-occurrence probabilities. In *Proceedings of the 32nd annual meeting of the ACL, Las Cruces, NM* (pp. 272–278).
- Daille, B. (1996). *Study and implementation of combined techniques for automatic extraction of terminology. The balancing act: Combining symbolic and statistical approaches to language*. Cambridge: MIT Press.
- Das, S. R., & Chen, M. Y. (2001). Yahoo! for Amazon: Sentiment parsing from small talk on the web. In *Proceedings of the 8th Asia Pacific finance association annual conference*.
- Dave, K., Lawrence, S., & Pennock, D. M. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *WWW2003, Budapest, Hungary*.
- Devitt, A., & Vogel, C. (2004). The topology of WordNet: Some metrics. In *Proceedings of GWC-04, 2nd global WordNet conference, Brno, CZ* (pp. 106–111).
- Didier, B. (1995). Lexter: A terminology extraction software for knowledge acquisition from texts. In *Proceedings 9th Banff knowledge acquisition for knowledge-based systems workshop, Banff* (Vol. 5, pp. 1–17).
- Dimitrova, M., Finn, A., Kushmerick, N., & Smyth, B. (2002). Web genre visualisation. In *Conference on human factors in computing systems*.
- Durbin, S. D., Neal Richter, J., & Warner, D. (2003). A system for affective rating of texts. In *Proceedings of OTC-03, 3rd workshop on operational text classification, Washington, US*.
- Edmonds, P. (1999). *Semantic representations of near-synonyms for automatic lexical choice*. Ph.D. thesis, University of Toronto.
- Edmonds, P., & Hirst, G. (2002). Near-synonymy and lexical choice. *Computational Linguistics*, 28, 105–144.
- Efron, M. (2004). Cultural orientation: Classifying subjective documents by collocation analysis. In *Proceedings of the AAAI fall symposium on style and meaning in language* (pp. 41–48). (Art, music, and design).
- Esuli, A., & Sebastiani, F. (2005). Determining the semantic orientation of terms through gloss classification. In *Proceedings of CIKM-05, the ACM SIGIR conference on information and knowledge management, Bremen, DE*.
- Esuli, A., & Sebastiani, F. (2006). SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC 2006, 5th conference on language resources and evaluation, Genova*.
- Esuli, A., & Sebastiani, F. (2006). Determining term subjectivity and term orientation for opinion mining. In *Proceedings of EACL-06, 11th conference of the european chapter of the association for computational linguistics, Trento*.
- Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A.-M., Shaked, T., et al. (2004). Web-scale information extraction in KnowItAll (preliminary results). In *WWW'2004*.
- Fei, Z., Liu, J., & Wu, G. (2004). Sentiment classification using phrase patterns. In *Proceedings of the fourth international conference on computer and information technology (CIT'04)*.
- Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.
- Finn, A., Kushmerick, N., & Smyth, B. (2002). Genre classification and domain transfer for information filtering. In F. Crestani, M. Girolami, C. van Rijsbergen, & J. Cornelis (Eds.), *Proceedings of ECIR-02, 24th European colloquium on information retrieval research, Glasgow, UK*. Heidelberg, DE: Springer-Verlag.
- Freitag, D., & McCallum, A. (2000). Information extraction with HMM structures learned by stochastic optimization. In *AAAI'00*.

- Gamon, M. (2004). Sentiment classification on customer feedback data: Noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings the 20th international conference on computational linguistics* (pp. 841–847).
- 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* (pp. 57–64).
- Glance, N. S., Hurst, M., & Tomokiyo, T. (2004). Blogpulse: Automated trend discovery for weblogs. In *WWW 2004 workshop on the weblogging ecosystem: Aggregation, analysis and dynamics*.
- Grefenstette, Gregory (1994). *Explorations in automatic thesaurus discovery*. Boston, MA: Kluwer Academic Press.
- 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 RIAO-04, 7th international conference on recherche d'information assistée par ordinateur, Avignon, FR* (pp. 186–194).
- Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 35th annual meeting of ACL*.
- Hatzivassiloglou, V., & Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *Proceedings of the 18th international conference on computational linguistics*.
- Hayakawa, S. I. (1994). *Choose the right word* (2nd ed.). Harper-Collins Publishers. (revised by Eugene Ehrlich).
- Hillard, D., Ostendorf, M., & Shriberg, E. (2003). Detection of agreement vs. disagreement in meetings: Training with unlabeled data. In *Proceedings of HLTNAACL*.
- Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. In *KDD'04, Seattle, Washington, USA*.
- Hu, M., & Liu, B. (2004). Mining opinion features in customer reviews. In *AAAI 2004* (pp. 755–760).
- Inkpen, D. Z., Feiguina, O., & Hirst, G. (2004). Generating more-positive and more-negative text. In J. G. Shanahan, Y. Qu, & J. Wiebe (Eds.), *Computing attitude and affect in text: Theory and applications. The information retrieval series* (Vol. 20, pp. 187–196). Dordrecht, The Netherlands: Springer.
- Jacquemin, C., & Bourigault, D. (2001). Term extraction and automatic indexing. In R. Mitkov (Ed.), *Handbook of computational linguistics*. Oxford University Press.
- Jon, K., & Tardos, E. (2002). Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields. *Journal of the ACM*, 49(5), 616–639.
- Justeson, J., & Slava, K. (1993). Technical terminology: Some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(1), 9–27.
- Kamps, J., & Marx, M. (2002). Words with attitude. In *Proceedings of the first international conference on global WordNet, CILL, Mysore, India* (pp. 332–341).
- Kamps, J., Marx, M., Mokken, R. J., & de Rijke, M. (2004). Using WordNet to measure semantic orientation of adjectives. In *Proceedings of LREC-04, 4th international conference on language resources and evaluation, Lisbon, PT* (Vol. IV, pp. 1115–1118).
- Kennedy, A., & Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2), 110–125.
- Kessler, B., Nunberg, G., & Schütze, H. (1997). Automatic detection of text genre. In *Proceedings of the 35th ACL/8th EACL* (pp. 32–38).
- Kim, S.-M., & Hovy, E. (2005). Automatic detection of opinion bearing words and sentences. In *Companion volume of the proceedings of IJCNLP-05, Jeju Island, Republic of Korea*.
- Kim, S.-M., Hovy, E. (2006). Identifying and analyzing judgment opinions. In *Proceedings of HLT/NAACL-2006, New York City*.
- Kim, S.-M., Hovy, E. (2006). Automatic identification of pro and con reasons in online reviews. In *Proceedings of the COLING/ACL 2006 main conference poster sessions, Sydney* (pp. 483–490).
- Kim, S.-M., & Hovy, E. (2004). Determining the sentiment of opinions. *Coling*, 1367–1373.
- Kobayashi, N., Inui, T., & Inui, K. (2001). Dictionary-based acquisition of the lexical knowledge for p/n analysis (in Japanese). In *Proceedings of Japanese society for artificial intelligence, SLUD-33* (pp. 45–50).
- Koppel, M., & Schler, J. (2005). The importance of neutral examples for learning sentiment. In *Workshop on the analysis of informal and formal information exchange during negotiations (FINEXIN)*.
- Koppel, M., Schler, J. (2005). Using neutral examples for learning polarity. In *Proceedings of IJCAI (poster)*.
- Ku, L.-W., Lee, L.-Y., Wu, T.-H., & Chen, H.-H. (2005). Major topic detection and its application to opinion summarization. In *SIGIR 2005* (pp. 627–628).
- Ku, L.-W., Wu, T.-H., Lee, L.-Y., & Chen, H.-H. (2005). Construction of an evaluation corpus for opinion extraction. In *Proceedings of NTCIR-5 workshop meeting, Tokyo, Japan*.
- Ku, L.-W., Liang, Y.-T., & Chen, H.-H. (2006). Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI-CAAW'06*.
- Lafferty, J., McCallum, A., & Pereira, F. (2001). Conditional random fields: Probabilistic models for segmenting and labeling or sequence data. In *ICML'01*.
- Leacock, C., & Chodorow, M. (1998). Combining local context and WordNet similarity for word sense identification. *WordNet: An electronic lexical database* (pp. 265–283). MIT Press: Cambridge MA.
- Leshed, G., & Kaye, J. (2006). Understanding how bloggers feel: Recognizing affect in blog posts. In *Extended Abstracts of CHI 2006* (pp. 1019–1024).
- Li, H., & Yamanishi, K. (2001). Mining from open answers in questionnaire data. In *Proceedings of the 7th ACM SIGKDD conference*.
- Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL* (pp. 768–774).
- 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 the 10th conference on computational natural language learning (CoNLL-X), New York City* (pp. 109–116).
- Liu, B., Hsu, W., & Ma, Y. (1998). Integrating classification and association rule mining. In *KDD'98*.
- Liu, H., Lieberman, H., & Selker, T. (2003). A model of textual affect sensing using real-world knowledge. In *IUI'03: Proceedings of the 8th international conference on intelligent user interfaces* (pp. 125–132). New York, NY, USA: ACM Press.
- Liu, B., Hu, M., & Cheng, J. (2005). Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th international world wide web conference (WWW-2005)* (pp. 10–14). Chiba, Japan: ACM Press.
- Losee, R. M. (2001). Natural language processing in support of decisionmaking: Phrases and part-of-speech tagging. *Information Processing and Management*, 37(6), 769–787.
- Luca, D., & Mazzini, G. (2002). Opinion classification through information extraction. In *International conference on data mining methods and databases for engineering, finance and other fields* (pp. 299–310).
- Matsumoto, S., Takamura, H., & Okumura, M. (2005). Sentiment classification using word sub-sequences and dependency sub-trees. In *PAKDD 2005* (pp. 301–311). Springer-Verlag: Berlin, Heidelberg.
- Mei, Q., Ling, X., Wondra, M., Su, H., & Zhai, C. X. (2007). Topic sentiment mixture: Modeling facets and opinions in weblogs. In *WWW 2007, Banff, Alberta, Canada*.
- Miller, G. A., & Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6, 1–28.
- Mishne, G., & de Rijke, M. (2006). Capturing global mood levels using blog posts. In *AAAI 2006 spring symposium on computational approaches to analysing weblogs (AAAI/CAAW 2006)*.
- Morinaga, S., Yamanishi, K., Tateishi, K., & Fukushima, T. (2002). Mining product reputations on the web. In *ACM SIGKDD 2002* (pp. 341–349).
- Mullen, T., & Collier, N. (2004). Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of EMNLP-2004, Barcelona, Spain* (pp. 412–418).
- Mullen, T., & Malouf, R. (2006). A preliminary investigation into sentiment analysis of informal political discourse. In *Proceedings of the AAAI symposium on computational approaches to analyzing weblogs* (pp. 159–162).
- Nasukawa, T., & Yi, J. (2003). Sentiment analysis: Capturing favorability using natural language processing. In *Second international conference on knowledge capture, Florida, USA*.
- Nigam, K., McCallum, A., & Thrun, S. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3), 103–134.
- Pang, Bo., & Lee, L. (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings 42nd ACL, Barcelona, Spain* (pp. 271–278).
- Pang, Bo., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 conference on empirical methods in natural language processing (EMNLP)* (pp. 79–86).
- Pereira, F. C. N., Tishby, N., & Lee, L. (1993). Distributional clustering of English words. In *Meeting of the association for computational linguistics* (pp. 183–190).
- Philip, B., Hastie, T., Christopher M., & Shivakumar V. (2004). Exploring sentiment summarization. In *AAAI spring symposium on exploring attitude and affect in text: Theories and applications (AAAI tech report SS-04-07)*.
- Rapp, R. (2004). A freely available automatically generated thesaurus of related words. In *Proceedings of 4th international conference on language resources and evaluation (LREC 2004), Lisbon, Portugal*.
- Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In *Proceedings of the EMNLP conference* (pp. 133–142).
- Rauber, A., & Muller-Kogler, A. (2001). Integrating automatic genre analysis into digital libraries. In *First ACM-IEEE joint conference on digital libraries*.
- Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *ACL student research workshop* (pp. 43–48). Ann Arbor, MI: Association for Computational Linguistics.
- Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th international joint conference on artificial intelligence*. Morgan Kaufmann.
- Riloff, E. (1996). An empirical study of automated dictionary construction for information extraction in three domains. In *Artificial intelligence* (Vol. 85).
- Riloff, E., & Jones, R. (1999). Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the 16th national conference on artificial intelligence*.
- Riloff, E., & Shepherd, J. (1997). A corpus-based approach for building semantic lexicons. In *Proceedings of the second conference on empirical methods in natural language processing* (pp. 117–124).
- Riloff, E., & Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on EMNLP* (pp. 105–112).
- Riloff, E., Wiebe, J., & Wilson, T. (2003). Learning subjective nouns using extraction pattern bootstrapping. In *Conference on natural language learning (CoNLL), Edmonton* (pp. 25–32).
- Roark, B., & Charniak, E. (1998). Noun-phrase co-occurrence statistics for semi-automatic semantic lexicon construction. In *Proceedings of the 36th annual meeting of the association for computational linguistics* (pp. 1110–1116).
- Rosario, B., & Hearst, M. A. (2004). Classifying semantic relations in bioscience text. In *ACL 2004*.

- Salveti, F., Lewis, S., & Reichenbach, C. (2004). Automatic opinion polarity classification of movie reviews. *Colorado research in linguistics* (Vol. 17, no. 1), Boulder: University of Colorado.
- Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International conference on new methods in language processing, Manchester, UK*.
- Spertus, E. (1997). Smokey: Automatic recognition of hostile messages. In *Proceedings IAAI*.
- Stone, P. J., Dunphy, D. C., Smith, M. S., & Ogilvie, D. M. (1966). *The general inquirer: A computer approach to content analysis*. Cambridge, MA: MIT Press.
- Stoyanov, V., Cardie, C., Litman, D., & Wiebe, J. (2004). Evaluating an opinion annotation scheme using a new multi-perspective question and answer corpus. In G. James, Y. Q. Shanahan, J. Wiebe (Eds.), *Computing attitude and affect in text: Theory and applications*. The information retrieval series, (Vol. 20, pp. 77–89). Dordrecht, The Netherlands: Springer.
- Subasic, P., & Huettner, A. (2001). Affect analysis of text using fuzzy typing, fuzzy systems. *IEEE Transactions*, 9, 483–496.
- Taboada, M., Caroline A., & Kimberly V. (2006). Creating semantic orientation dictionaries. In *Proceedings of 5th international conference on language resources and evaluation (LREC), Genoa, Italy*.
- Taboada, M., Gillies, M. A., & McFetridge, P. (2006). Sentiment classification techniques for tracking literary reputation. In *Proceedings of LREC 2006 workshop "Towards Computational Models of Literary Analysis"* (pp. 36–43).
- Takamura, H., Inui, T., & Okumura, M. (2005). Extracting semantic orientations of words using spin model. In *Proceedings of the 43rd annual meeting of the ACL, Ann Arbor* (pp. 133–140).
- Tan, S., Wu, G., Tang, H., & Cheng, X. (2007). A novel scheme for domain-transfer problem in the context of sentiment analysis. In *Proceedings of CIKM'07, Lisboa, Portugal*.
- Terveen, L., Hill, W., Amento, B., McDonald, D., & Creter, J. (1997). PHOAKS: A system for sharing recommendations. *Communications of the ACM*, 40(3), 59–62.
- Thelen, M., & Riloff, E. (2002). A bootstrapping method for learning semantic lexicons using extraction pattern contexts. In *Proceedings of the 2002 conference on empirical methods in natural language processing*.
- Thomas, M., Pang, Bo., & Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 conference on empirical methods in natural language processing (EMNLP 2006), Sydney* (pp. 327–335).
- Tong, R. M. (2001). An operational system for detecting and tracking opinions in on-line discussion. In *SIGIR 2001 workshop on operational text classification*.
- Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting of the association for computational linguistics (ACL), Philadelphia* (pp. 417–424).
- Turney, P. D., & Littman, M. L. (2002). *Unsupervised learning of semantic orientation from a hundred-billion-word corpus*. Technical Report ERB-1094. National Research Council Canada, Institute for Information Technology.
- Turney, P. D., & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems*, 21(4), 315–346.
- Valitutti, A., Strapparava, C., & Stock, O. (2004). Developing affective lexical resources. *Psychology Journal*, 2(1), 61–83.
- Vegnaduzzo, S. (2004). Acquisition of subjective adjectives with limited resources. In *Proceedings of the AAAI spring symposium on exploring attitude and affect in text: Theories and applications, Stanford, US*.
- Whitelaw, C., Argamon, S., & Garg, N. (2005). Using appraisal taxonomies for sentiment analysis. In *Proceedings of the first computational systemic functional grammar conference, University of Sydney, Sydney, Australia*.
- Whitelaw, C., Garg, N., & Argamon, S. (2005). Using appraisal groups for sentiment analysis. In *Proceedings of CIKM-05, 14th ACM international conference on information and knowledge management, Bremen, DE* (pp. 625–631).
- Wiebe, J. (1994). Tracking point of view in narrative. *Computational Linguistics*, 20(2), 233–287.
- Wiebe, J. (2000). Learning subjective adjectives from corpora. In *AAAI/IAAI* (pp. 735–740).
- Wiebe, J., & Riloff, E. (2005). Creating subjective and objective sentence classifiers from unannotated texts. In *Sixth international conference on intelligent text processing and computational linguistics*.
- Wiebe, J., Bruce, R., & O'Hara, T. (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 (ACL-99)* (pp. 246–253).
- Wiebe, J., Wilson, T., Bruce, R., Bell, M., & Martin, M. (2002). *Learning subjective language*. Technical Report TR-02-100. Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania.
- Wiebe, J., Wilson, T., & Bell, M. (2001). Identifying collocations for recognizing opinions. In *Proceedings of the ACL/EACL workshop on collocation*. ACL.
- Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 1(2).
- Wilks, Y., & Stevenson, M. (1998). The grammar of sense: Using part-of-speech tags as a first step in semantic disambiguation. *Journal of Natural Language Engineering*, 4(2), 135–144.
- Wilson, T., Wiebe, J., & Hwa, R. (2004). Just how mad are you? Finding strong and weak opinion clauses. In *Proceedings of AAAI-04, 21st conference of the american association for artificial intelligence, San Jose, US*.
- Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technologies conference/conference on empirical methods in natural language processing (HLT/EMNLP 2005), Vancouver, Canada*.
- Ye, Q., Shi, W., & Li, Y. (2006). Sentiment classification for movie reviews in Chinese by improved semantic oriented approach. In *Proceedings of the 39th Hawaii international conference on system sciences*.
- 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 third IEEE international conference on data mining* (p. 427).
- Yu, H., & Hatzivassiloglou, V. (2003). Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on empirical methods in natural language processing* (pp. 129–136).
- Zadeh, L. A. (1987). PRUF—a meaning representation language for natural languages. In R. R. Yager, S. Ovchinnikov, R. M. Tong, & H. T. Nguyen (Eds.), *Fuzzy sets and applications* (pp. 499–568). John Wiley & Sons.
- Zhai, Y., & Liu B. (2005). Web data extraction based on partial tree alignment. In *WWW'05*.
- Zhang, Y., Xu, W., & Callan, J. (2002). Exact maximum likelihood estimation for word mixtures. In *ICML workshop on text learning*.