Sentiment analysis and subjectivity

Bing Liu
University of Illinois at Chicago

To appear in ***Handbook of Natural Language Processing***, Second Edition, (editors: N. Indurkhya and F. J. Damerau), 2010

Sentiment Analysis and Subjectivity

Bing Liu
Department of Computer Science
University of Illinois at Chicago
liub@cs.uic.edu

Textual information in the world can be broadly categorized into two main types: *facts* and *opinions*. Facts are objective expressions about entities, events and their properties. Opinions are usually subjective expressions that describe people's sentiments, appraisals or feelings toward entities, events and their properties. The concept of opinion is very broad. In this chapter, we only focus on opinion expressions that convey people's positive or negative sentiments. Much of the existing research on textual information processing has been focused on mining and retrieval of factual information, e.g., information retrieval, Web search, text classification, text clustering and many other text mining and natural language processing tasks. Little work had been done on the processing of opinions until only recently. Yet, opinions are so important that whenever we need to make a decision we want to hear others' opinions. This is not only true for individuals but also true for organizations.

One of the main reasons for the lack of study on opinions is the fact that there was little opinionated text available before the World Wide Web. Before the Web, when an individual needed to make a decision, he/she typically asked for opinions from friends and families. When an organization wanted to find the opinions or sentiments of the general public about its products and services, it conducted opinion polls, surveys, and focus groups. However, with the Web, especially with the explosive growth of the user-generated content on the Web in the past few years, the world has been transformed.

The Web has dramatically changed the way that people express their views and opinions. They can now post reviews of products at merchant sites and express their views on almost anything in Internet forums, discussion groups, and blogs, which are collectively called the *user-generated content*. This online word-of-mouth behavior represents new and measurable sources of information with many practical applications. Now if one wants to buy a product, he/she is no longer limited to asking his/her friends and families because there are many product reviews on the Web which give opinions of existing users of the product. For a company, it may no longer be necessary to conduct surveys, organize focus groups or employ external consultants in order to find consumer opinions about its products and those of its competitors because the user-generated content on the Web can already give them such information.

However, finding opinion sources and monitoring them on the Web can still be a formidable task because there are a large number of diverse sources, and each source may also have a huge volume of *opinionated text* (text with opinions or sentiments). In many cases, opinions are hidden in long forum posts and blogs. It is difficult for a human reader to find relevant sources, extract related sentences with opinions, read them, summarize them, and organize them into usable forms. Thus, automated opinion discovery and summarization systems are needed. *Sentiment analysis*, also known as *opinion mining*, grows out of this need. It is a challenging natural language processing or text mining problem. Due to its tremendous value for practical applications, there has been an explosive growth of both research in academia and applications in the industry. There are now at least 20-30 companies that offer sentiment analysis services in USA alone. This chapter introduces this research field. It focuses on the following topics:

1. **The problem of sentiment analysis**: As for any scientific problem, before solving it we need to define or to formalize the problem. The formulation will introduce the basic definitions, core concepts and issues, sub-problems and target objectives. It also serves as a common framework to unify different research directions. From an application point of view, it tells practitioners what the main tasks are, their inputs and outputs, and how the resulting outputs may be used in practice.
2. **Sentiment and subjectivity classification:** This is the area that has been researched the most in academia. It treats sentiment analysis as a text classification problem. Two sub-topics that have been

extensively studied are: (1) classifying an opinionated document as expressing a positive or negative opinion, and (2) classifying a sentence or a clause of the sentence as subjective or objective, and for a subjective sentence or clause classifying it as expressing a positive, negative or neutral opinion. The first topic, commonly known as *sentiment classification* or *document-level sentiment classification*, aims to find the general sentiment of the author in an opinionated text. For example, given a product review, it determines whether the reviewer is positive or negative about the product. The second topic goes to individual sentences to determine whether a sentence expresses an opinion or not (often called *subjectivity classification*), and if so, whether the opinion is positive or negative (called *sentence-level sentiment classification*).

3. **Feature-based sentiment analysis:** This model first discovers the targets on which opinions have been expressed in a sentence, and then determines whether the opinions are positive, negative or neutral. The targets are objects, and their components, attributes and features. An object can be a product, service, individual, organization, event, topic, etc. For instance, in a product review sentence, it identifies product features that have been commented on by the reviewer and determines whether the comments are positive or negative. For example, in the sentence, "*The battery life of this camera is too short*," the comment is on "battery life" of the camera object and the opinion is negative. Many real-life applications require this level of detailed analysis because in order to make product improvements one needs to know what components and/or features of the product are liked and disliked by consumers. Such information is not discovered by sentiment and subjectivity classification.
4. **Sentiment analysis of comparative sentences:** Evaluation of an object can be done in two main ways, direct appraisal and comparison. Direct appraisal, called *direct opinion*, gives positive or negative opinion about the object without mentioning any other similar objects. Comparison means to compare the object with some other similar objects (e.g., competing products). For example, "*The picture quality of this camera is poor*" expresses a direct opinion, while "*The picture quality of this camera is better than that of Camera-x*." expresses a comparison. Clearly, it is useful to identify such sentences, extract comparative opinions expressed in them and determine which objects are preferred by the sentence authors (in the above example, Camera-x is preferred with respect to the picture quality).
5. **Opinion search and retrieval:** Since the general Web search has been so successful in many aspects, it is not hard to imagine that opinion search will be very useful as well. For example, given a keyword query "gay marriage", one wants to find positive and negative opinions on the issue from an opinion search engine. For such a query, two tasks need to be performed: (1) retrieving documents or sentences that are relevant to the query, and (2) identifying and ranking opinionated documents or sentences from these retrieved. Opinion search is thus a combination of information retrieval and sentiment analysis.
6. **Opinion spam and utility of opinions:** As opinions on the Web are important for many applications, it is no surprise that people have started to game the system. Opinion spam refers to fake or bogus opinions that try to deliberately mislead readers or automated systems by giving undeserving positive opinions to some target objects in order to promote the objects and/or by giving malicious negative opinions to some other objects in order to damage their reputations. Detecting such spam is very important for applications. The utility of opinions refers to the usefulness or quality of opinions. Automatically assigning utility values to opinions is useful as opinions can then be ranked based on their utility values. With the ranking, the reader can focus on those quality opinions. We should note, however, that spam and utility are very different concepts, as we will see later.

In [72], Pang and Lee wrote a comprehensive survey of the sentiment analysis and opinion mining research. This chapter is not meant to be another such survey, but instead to introduce the field for teaching and learning. It focuses on the core topics of the research that are also essential for practical applications. It introduces the topics in sufficient detail so that the reader can have a good understanding of the main ideas without referring to the original papers. Another key characteristic of this chapter is that it takes a structured approach to exploring the problem. In non-NLP literature, natural language documents are regarded as unstructured data, while the data in relational databases are referred to as structured data. The structured approach means to turn unstructured text to structured data, which enables traditional data management tools to be applied to slice, dice, and visualize the results in many ways. This is extremely important for applications because it allows the user to gain insights through both qualitative and quantitative analysis.

1. The Problem of Sentiment Analysis

Sentiment analysis or opinion mining is the computational study of opinions, sentiments and emotions expressed in text. We use the following review segment on iPhone to introduce the problem (an number is associated with each sentence for easy reference):

> "(1) *I bought an iPhone a few days ago.* (2) *It was such a nice phone.* (3) *The touch screen was really cool.* (4) *The voice quality was clear too.* (5) *Although the battery life was not long, that is ok for me.* (6) *However, my mother was mad with me as I did not tell her before I bought it.* (7) *She also thought the phone was too expensive, and wanted me to return it to the shop. …* "

The question is: what we want to mine or extract from this review? The first thing that we may notice is that there are several opinions in this review. Sentences (2), (3) and (4) express positive opinions, while sentences (5), (6) and (7) express negative opinions or emotions. Then we also notice that the opinions all have some targets or objects on which the opinions are expressed. The opinion in sentence (2) is on the iPhone as a whole, and the opinions in sentences (3), (4) and (5) are on the "touch screen", "voice quality" and "battery life" of the iPhone respectively. The opinion in sentence (7) is on the price of the iPhone, but the opinion/emotion in sentence (6) is on "me", not iPhone. This is an important point. In an application, the user may be interested in opinions on certain targets or objects, but not on all (e.g., unlikely on "me"). Finally, we may also notice the sources or holders of opinions. The source or holder of the opinions in sentences (2), (3), (4) and (5) is the author of the review ("I"), but in sentences (6) and (7) is "my mother". With this example in mind, we now formally define the sentiment analysis or opinion mining problem. We start with the opinion target.

In general, opinions can be expressed on anything, e.g., a product, a service, an individual, an organization, an event, or a topic. We use the term *object* to denote the target entity that has been commented on. An object can have a set of *components* (or *parts*) and a set of *attributes* (or *properties*). Each component may have its own sub-components and its set of attributes, and so on. Thus, an object can be hierarchically decomposed based on the *part-of* relation. Formally, we have the following [55]:

Definition (**object**)**:** An *object o* is an entity which can be a product, person, event, organization, or topic. It is associated with a pair, o: (T, A), where T is a hierarchy of *components* (or *parts*), *sub-components*, and so on, and A is a set of *attributes* of o. Each component has its own set of sub-components and attributes.

Example 1: A particular brand of cellular phone is an object. It has a set of components, e.g., *battery*, and *screen*, and also a set of attributes, e.g., *voice quality*, *size*, and *weight*. The battery component also has its set of attributes, e.g., *battery life*, and *battery size*.

Based on this definition, an object can be represented as a tree, hierarchy or taxonomy. The root of the tree is the object itself. Each non-root node is a component or sub-component of the object. Each link is a *part-of* relation. Each node is also associated with a set of attributes or properties. An opinion can be expressed on any node and any attribute of the node.

Example 2: Following Example 1, one can express an opinion on the cellular phone itself (the root node), e.g., "*I do not like this phone*", or on one of its attributes, e.g., "*The voice quality of this phone is lousy*". Likewise, one can also express an opinion on any one of the phone's components or any attribute of the component.

In practice, it is often useful to simplify this definition due to two reasons: First, natural language processing is a difficult task. To effectively study the text at an arbitrary level of detail as described in the definition is extremely challenging. Second, for an ordinary user, it is probably too complex to use a hierarchical representation of an object and opinions on the object. Thus, we flatten the tree to omit the hierarchy and use the term *features* to represent both components and attributes. In this simplification, the object itself can also be seen as a feature (but a special feature), which is the root of the original tree. An opinionated comment on the object itself is called a *general opinion* on the object (e.g., "*I like iPhone*"). An opinionated comment on any specific feature is called a *specific opinion* on a feature of the object, e.g., "*The touch screen of iPhone is really cool*", where "touch screen" is a feature of iPhone.

Using features for an object is quite common in the product domain as people often use the term *product features*. However, when the objects are events and topics, the term *feature* may not sound natural. Indeed in some other domains, researchers also use the term *topic* [46] or *aspect* [50, 84] to mean *feature*. In this chapter, we choose to use the term *feature* along with the term *object*. We should note that both terms are needed because in most applications the primary concern of the user is a set of objects of interest (e.g., a set of competing products). Then we need to know each feature talked about in an opinion document belonging to which object. One issue with the term feature is that it can confuse with the term feature used in machine learning, where a feature means a data attribute. To avoid the confusion, we will use the term *object feature* to mean feature of an object whenever such confusion may arise.

Let an *opinionated document* be d, which can be a product review, a forum post or a blog that evaluates a set of objects. In the most general case, d consists of a sequence of sentences $d = \langle s_1, s_2, \ldots, s_m \rangle$.

Definition (**opinion passage on a feature**)**:** An *opinion passage* on a feature f of an object O evaluated in d is a group of consecutive sentences in d that expresses a positive or negative opinion on f.

It is possible that a sequence of sentences (at least one) in an opinionated document together expresses an opinion on an object or a feature of the object. It is also possible that a single sentence expresses opinions on more than one feature, e.g.,

"The voice quality of this phone is good, but the battery life is short".

Much of the current research focuses on sentences, i.e., each passage consisting of a single sentence. In the subsequent discussion, we also treat each sentence as the basic information unit.

Definition (**explicit and implicit feature**)**:** If a feature f or any of its synonyms appears in a sentence s, f is called an *explicit feature* in s. If neither f nor any of its synonyms appear in s but f is implied, then f is called an *implicit feature* in s.

Example 3: "battery life" in the following sentence is an explicit feature:

"The battery life of this phone is too short".

Size is an implicit feature in the following sentence as it does not appear in the sentence but it is implied:

"This phone is too large".

Here, "large", which is not a synonym of *size*, is called a *feature indicator*. Many feature indicators are adjectives and adverbs. Some adjectives and adverbs are general and can be used to modify anything, e.g., *good*, *bad*, and *great*, but many actually indicate the types of features that they are likely to modify, e.g., *beautiful* (appearance), and *reliably* (reliability). Thus, such feature indicators may be directly mapped to their underlying features. We will discuss this again in Section 3.1.2.

Definition (**opinion holder**)**:** The *holder* of an opinion is the person or organization that expresses the opinion.

Opinion holders are also called *opinion sources* [101]. In the case of product reviews and blogs, opinion holders are usually the authors of the posts. Opinion holders are more important in news articles because they often explicitly state the person or organization that holds a particular opinion [5, 14, 46]. For example, the opinion holder in the sentence "*John expressed his disagreement on the treaty*" is "John".

Definition (**opinion**): An *opinion* on a feature f is a positive or negative view, attitude, emotion or appraisal on f from an opinion holder.

Definition (**opinion orientation**)**:** The *orientation* of an opinion on a feature f indicates whether the opinion is *positive*, *negative* or *neutral*.

Opinion orientation is also known as *sentiment orientation*, *polarity of opinion*, or *semantic orientation.*

We now put everything together to define a model of an object, a model of an opinionated text, and the mining objective, which are collectively called the *feature-based sentiment analysis model* [36, 55, 56].

Model of an object: An object o is represented with a finite set of features, $F = \{f_1, f_2, \ldots, f_n\}$, which includes the object itself as a special feature. Each feature $f_i \in F$ can be expressed with any one of a finite set of words or phrases $W_i = \{w_{i1}, w_{i2}, \ldots, w_{im}\}$, which are *synonyms* of the feature, or indicated by any one of a finite set of feature indicators $I_i = \{i_{i1}, i_{i2}, \ldots, i_{iq}\}$ of the feature.

Model of an opinionated document: A general opinionated document d contains opinions on a set of objects $\{o_1, o_2, \ldots, o_q\}$ from a set of opinion holders $\{h_1, h_2, \ldots, h_p\}$. The opinions on each object o_j are expressed on a subset F_j of features of o_j. An opinion can be any one of the following two types:

1. **Direct opinion**: A *direct opinion* is a quintuple (o_j, f_{jk}, oo_{ijkl}, h_i, t_l), where o_j is an object, f_{jk} is a feature of the object o_j, oo_{ijkl} is the orientation or polarity of the opinion on feature f_{jk} of object o_j, h_i is the opinion holder and t_l is the time when the opinion is expressed by h_i. The opinion orientation oo_{ijkl} can be positive, negative or neutral (or measured based on a more granular scale to express different strengths of opinions [103]). For feature f_{jk} that opinion holder h_i comments on, he/she chooses a word or phrase from the corresponding synonym set W_{jk}, or a word or phrase from the corresponding feature indicator set I_{jk} to describe the feature, and then expresses a positive, negative or neutral opinion on the feature.

2. **Comparative opinion**: A *comparative opinion* expresses a relation of similarities or differences between two or more objects, and/or object preferences of the opinion holder based on some of the shared features of the objects. A comparative opinion is usually expressed using the *comparative* or *superlative* form of an adjective or adverb, although not always. More detailed discussions will be given in Section 4. The discussion below focuses only on direct opinions.

This opinionated text model covers the essential but not all the interesting information or all possible cases. For example, it does not cover the situation described in the following sentence: "*The view-finder and the lens of this camera are too close*", which expresses a negative opinion on the distance of the two components. We will follow this simplified model in the rest of this chapter as it is often sufficient for practical applications.

On direct opinions, there are in fact two main sub-types. In the first sub-type, opinions are directly expresses on an object or features of the object, e.g., "*The voice quality of this phone is great*." In the second sub-type, opinions on an object are expressed based on its effect on some other objects. This sub-type often occurs in the medical domain when patients express opinions on drugs or describe their side effects. For example, the sentence "*After taking this drug, my left knee felt great*" describes a desirable effect of the drug on the knee, and thus implies a positive opinion on the drug. We call both types direct opinions in this chapter for the sake of simplicity and to distinguish them from comparative opinions.

Before going further, let us also have some more discussions about the strength of an opinion (oo_{ijkl}). Opinions come in different strengths [103]. Some are very strong, e.g., "*This phone is a piece of junk*" and some are weak, e.g., "*I think this phone is fine*". Hence, the strength of opinions can be interpreted as scaled. For example, a positive opinion may express a feeling of *contented*, *happy*, *joyous*, or *ecstatic*, from the low intensity value of *contented* to the maximally high intensity value of *ecstatic* [61]. In a practical application, we can choose the number of strength values or levels depending on the application need. For example, for positive opinions, we may only need two levels, i.e., grouping *contented* and *happy* into one level, and *joyous* and *ecstatic* into the other level. This discussion in fact touches the concept of emotions.

Definition (emotions): Emotions are our subjective feelings and thoughts.

Emotions have been studied in many fields, e.g., psychology, philosophy, sociology, biology, etc. However, there is still not a set of agreed basic emotions of people among researchers. Based on [75], people have 6 types of primary emotions, i.e., *love*, *joy*, *surprise*, *anger*, *sadness* and *fear*, which can be sub-divided into many secondary and tertiary emotions. Each emotion can also have different intensities. The strengths of opinions are closely related to the intensities of certain emotions, e.g., joy and anger. However, the concepts of emotions and opinions are not equivalent although they have a large intersection.

When discussing subjective feelings of emotions or opinions, it is useful to distinguish two different notions: people's mental states (or feelings) and language expressions used to describe the mental states. Although there are only 6 types of emotions, there are a large number of language expressions that can be used to express them. Similarly, there are also a large (seemly unlimited) number of opinion expressions that describe positive or negative sentiments. Sentiment analysis or opinion mining essentially tries to infer people's sentiments based on their language expressions.

We now describe the objective of sentiment analysis or opinion mining, which not only aims to infer positive or negative opinions/sentiments from text, but also to discover the other pieces of associated information which are important for practical applications of the opinions.

Objective of mining direct opinions: Given an opinionated document d,

1. discover all opinion quintuples (o_j, f_{jk}, oo_{ijkl}, h_i, t_l) in d, and
2. identify all the synonyms (W_{jk}) and feature indicators I_{jk} of each feature f_{jk} in d.

Some remarks about this feature-based sentiment analysis or opinion mining model are as follows:

1. It should be stressed that the five pieces of information in the quintuple need to correspond to one another. That is, the opinion oo_{ijkl} must be given by opinion holder h_i on feature f_{jk} of object o_j at time t_l. This requirement gives some clue why sentiment analysis is such a challenging problem because even identifying each piece of information itself is already very difficult, let alone finding all five and match them. To make matters worse, a sentence may not explicitly mention some pieces of information, but they are implied due to pronouns, language conventions, and the context. Let us see an example blog (the number before each sentence is added as the sentence id to facilitate the discussion below):

   Example 4: "(1) *This past Saturday, I bought a Nokia phone and my girlfriend bought a Motorola phone.* (2) *We called each other when we got home.* (3) *The voice on my phone was not so clear, worse than my previous phone.* (4) *The camera was good.* (5) *My girlfriend was quite happy with her phone.* (6) *I wanted a phone with good voice quality.* (7) *So my purchase was a real disappointment.* (8) *I returned the phone yesterday*."

   The objects to be discovered in this blog are "Motorola phone" and "Nokia phone", which are by no means easy to identify in practice. To figure out what is "my phone" and what is "her phone" in sentences (3) and (5) is even more challenging. Sentence (4) does not mention any phone and does not have a pronoun. Then the question is which phone "the camera" belongs to. Sentence (6) seemingly expresses a positive opinion about a phone and its voice quality, but of course that is not the case. In sentences (7) and (8), it is hard to know what "my purchase" is and what "the phone" is. The opinion holder of all the opinions is the author of the blog except sentence (5) whose opinion holder is "my girlfriend."

2. In practice not all five pieces of information in the quintuple needs to be discovered for every application because some of them may be known or not needed. For example, in the context of product reviews, the object (product) evaluated in each review, the time when the review is submitted, and the opinion holder are all known as a review site typically records and displays such information. Of course, one still needs to extract such information from the Web page, which is usually a structured data extraction problem (see Chapter 9 of [55]).

Example 4 above revealed another issue, namely, *subjectivity*. That is, in a typical document (even an opinionated document), some sentences express opinions and some do not. For example, sentences (1), (2), (6) and (8) do not express any opinions. The issue of subjectivity has been extensively studied in the literature [34, 35, 79, 80, 97, 99, 100, 102, 103, 104].

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

For example, in Example 4, sentences (1), (2) and (8) are objective sentences, while all other sentences are subjective sentences. Subjective expressions come in many forms, e.g., opinions, allegations, desires, beliefs, suspicions, and speculations [79, 97]. Thus, a subjective sentence may not contain an opinion. For example, sentence (6) in Example 4 is subjective but it does not express a positive or negative opinion on any specific phone. Similarly, we should also note that not every objective sentence contains no opinion as the second sentence in Example 5 below shows.

Definition (**explicit and implicit opinion**)**:** An *explicit opinion* on feature *f* is an opinion explicitly expressed on *f* in a subjective sentence. An *implicit opinion* on feature *f* is an opinion on *f* implied in an objective sentence.

Example 5: The following sentence expresses an explicit positive opinion:

"The voice quality of this phone is amazing."

The following sentence expresses an implicit negative opinion:

"The earphone broke in two days."

Although this sentence states an objective fact, it implicitly indicates a negative opinion on the earphone. In fact, sentence (8) in Example 4 can also be said to imply a negative opinion. In general, objective sentences that imply positive or negative opinions often state the reasons for the opinions.

Definition (**opinionated sentence**): An *opinionated sentence* is a sentence that expresses explicit or implicit positive or negative opinions. It can be a subjective or objective sentence.

As we can see, the concepts of subjective sentences and opinionated sentences are not the same, although opinionated sentences are often a subset of subjective sentences. The approaches for identifying them are similar. Thus for simplicity of presentation, this chapter uses the two terms interchangeably. The task of determining whether a sentence is subjective or objective is called *subjectivity classification*.

Clearly, the idea of opinionated can also be applied to documents. So far we have taken opinionated documents for granted in the above definitions. In practice, they may also need to be identified. For example, many forum posts are questions and answers with no opinions. It is reasonable to say that whether a document is opinionated depends entirely on whether some of its sentences are opinionated. Thus, we may define a document to be opinionated if any of its sentences is opinionated. This definition, however, may not be suitable for all cases. For example, an objective news report may quote someone's opinion. It does not make good sense to say that the report is subjective or opinionated. It is perhaps more appropriate to say that the report contains some opinions. A more fair definition may be one that is based on the author's intension, i.e., whether he/she intends to express opinions on something using the text. Product reviews fit this definition, i.e., they are opinionated. Whether a sentence is opinionated or not is more clear-cut. In a typical document, some sentences are opinionated and some are not.

With the abstract model and mining objectives defined, we now see how the mining results may be presented to the user in applications. Although this step is not so much of academic research, it is crucial to applications. It also gives us some gleams of how an industrial user wants to see the results, which in turn also motivates our research. What we discuss below has already been used in the industry.

To start, we should note that for most opinion based applications, it is important to study a collection of opinions rather than only one because one opinion only represents the subjective view of a single person, *Cellular phone* 1:

PHONE:

Positive: 125 <individual review sentences>

Negative: 7 <individual review sentences>

Feature: **voice quality**

Positive: 120 <individual review sentences>

Negative: 8 <individual review sentences>

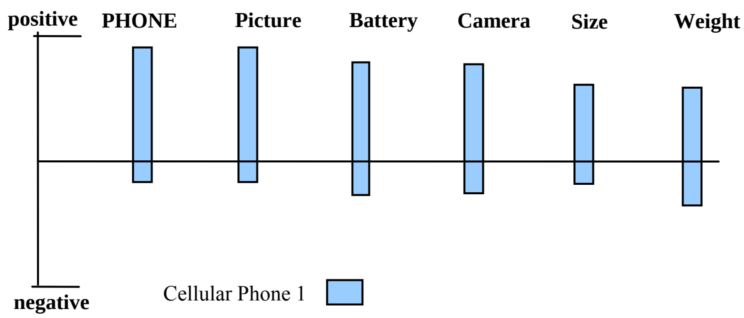
Feature: **size**

Positive: 80 <individual review sentences>

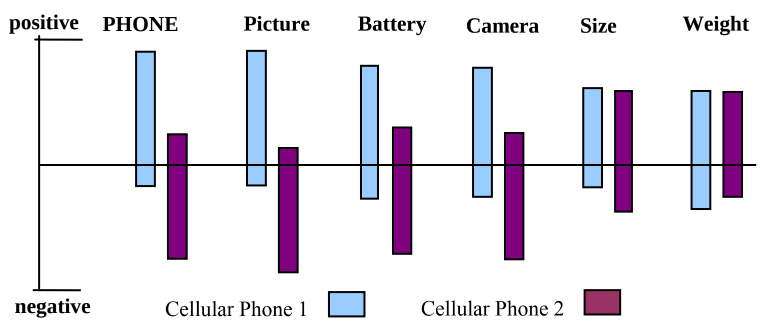
Negative: 12 <individual review sentences>

…

Figure 1. An example of a feature-based summary of opinions.



(A) Visualization of feature-based summary of opinions on a cellular phone



(B) Visual opinion comparison of two cellular phones

Figure 2. Visualization of feature-based summaries of opinions

which is usually not significant for action. This clearly indicates that some form of summary of the mining results is needed because it does not make sense to list all quintuples (opinions) to the user. Below, we use product reviews as an example to present some ideas.

Recall we mentioned at the beginning of the chapter that we wanted to turn unstructured natural language texts to structured data. The quintuple output does exactly that. All the discovered quintuples can be easily stored in database tables. A whole suite of database and visualization tools can then be applied to view the results in all kinds of ways to gain insights of consumer opinions, which are usually called *structured summaries* and are visualized as bar charts and/or pie charts.

Structured opinion summary: A simple way to use the results is to produce a *feature-based summary* of opinions on an object or multiple competing objects [36, 56].

Example 6: Assume we summarize the reviews of a particular cellular phone, *cellular phone* 1. The summary looks like that in Figure 1, which was proposed by Hu and Liu [36]. In the figure, "PHONE" represents the phone itself (the root node of the object hierarchy). 125 reviews expressed positive opinions on the phone and 7 reviews expressed negative opinions on the phone. “voice quality” and “size” are two product features. 120 reviews expressed positive opinions on the voice quality, and only 8 reviews expressed negative opinions. The <individual review sentences> link points to the specific sentences and/or the whole reviews that give positive or negative comments about the feature. With such a summary, the user can easily see how existing customers feel about the cellular phone. If he/she is interested in a particular feature, he/she can drill down by following the <individual review sentences> link to see why existing customers like it and/or what they are not satisfied with.

Such a summary can also be visualized easily using a bar chart [56]. Figure 2(A) shows the summary of opinions in a set of reviews of a cellular phone. In the figure, each bar above the *X*-axis in the middle shows the number of positive opinions on a feature (given at the top), and the bar below the *X*-axis shows the number of negative opinions on the same feature. Obviously, other similar visualizations are also possible. For example, we may only show the percent of positive opinions (the percent of negative opinions is just one minus the percent of positive opinions) for each feature. To see the actual review sentences behind each bar, the bar can be programmed in such a way that clicking on the bar will show all the review sentences in a popup window.

Comparing opinion summaries of a few competing products is even more interesting [56]. Figure 2(B) shows a visual comparison of consumer opinions on two competing phones. We can clearly see how consumers view different features of each product. Cellular phone 1 is definitely superior to cellular phone 2. Most customers have negative opinions about the voice quality, battery and camera of cellular phone 2. However, on the same three features, customers are mostly positive about cellular phone 1. Regarding the size and the weight, customers have similar opinions about both phones. For the phone itself (“PHONE”), most people are positive about cellular phone 1, but negative about cellular phone 2. Hence, the visualization enables users to see how the phones compare with each other along different feature dimensions.

Clearly, many other types of visualizations are possible, see [72] for a survey of other techniques. Incidentally, opinion summary of product reviews in Microsoft Bing search uses a bar chart similar to the one in Figure 2(A). At the time when this chapter was written, it did not provide the facility for side-by-side opinion comparison of different products as in Figure 2(B).

In fact, many types of summaries without opinions are also useful. We give some examples below.

Feature buzz summary: This summary shows the relative frequency of feature mentions. It can tell a company what their customers really care about. For example, in an online banking study, the most mentioned feature may be the transaction security.

Object buzz summary: This summary shows the frequency of mentions of different competing products. This is useful because it tells the popularity of different products or brands in the market place.

Since the time of the opinion is recorded in each quintuple, we can easily monitor changes of every aspect using trend tracking.

Trend tracking: If the time dimension is added to the above summaries, we get their trend reports. These reports can be extremely helpful in practice because the user always wants to know how things change over time [94].

All these summaries can be produced and visualized easily as they are just the results of some database queries with no additional mining. This shows the power of the structured output of opinion quintuples.

Finally, we note that researchers have also studied the summarization of opinions in the tradition fashion, e.g., producing a short textual summary based on multiple reviews or even a single review [4, 9, 52, 83, 88]. Such a summary gives the reader a quick overview of what people think about a product or service. However, one weakness of such a text-based summary is that it is often not quantitative but only qualitative, which is not suitable for analytical purposes, although it may be suitable for human reading. For example, a traditional text summary may say “*Most people do not like this product*”. However, a quantitative summary may say that 60% of the people do not like this product and 40% of them like it. In most opinion analysis applications, the quantitative aspect is crucial just like in the traditional survey research (in fact, reviews can be seen as open-ended surveys). In the survey research, structured summaries displayed as bar charts and pie charts are the most common approaches because they give the user a concise, quantitative and visual view.

Note that instead of generating a text summary directly from input reviews, it is also possible to generate a text summary based on the mining results as displayed in Figures 1 and 2. For example, it is easy to generate some natural language summary sentences based on what is shown on the bar chart using some predefined templates. For instance, the first two bars in Figure 2(B) can be summarized as "*Most people are positive about cellular phone 1 and negative about cellular phone 2*."

2. Sentiment and Subjectivity Classification

We now discuss some key research topics of sentiment analysis. *Sentiment classification* is perhaps the most widely studied topic [3, 6, 8, 12, 13, 15, 16, 18, 27, 28, 33, 34, 35, 44, 45, 62, 64, 66, 67, 68, 70, 71, 73, 79, 80, 86, 92, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 111]. It classifies an opinionated document (e.g., a product review) as expressing a positive or negative opinion. The task is also commonly known as the *document-level sentiment classification* because it considers the whole document as the basic information unit. The existing research assumes that the document is known to be opinionated. Naturally the same sentiment classification can also be applied to individual sentences. However, here each sentence is not assumed to be opinionated in the literature. The task of classifying a sentence as opinionated or not opinionated is called *subjectivity classification*. The resulting opinionated sentences are also classified as expressing positive or negative opinions, which is called the *sentence-level sentiment classification*.

2.1 Document-Level Sentiment Classification

Given a set of opinionated documents D, it determines whether each document $d \in D$ expresses a positive or negative opinion (or sentiment) on an object. Formally,

Task: Given an opinionated document d which comments on an object o, determine the orientation oo of the opinion expressed on o, i.e., discover the opinion orientation oo on feature f in the quintuple (o, f, so, h, t), where $f = o$ and h, t, o are assumed to be known or irrelevant.

Existing research on sentiment classification makes the following assumption:

Assumption: The opinionated document d (e.g., a product review) expresses opinions on a single object o and the opinions are from a single opinion holder h.

This assumption holds for customer reviews of products and services. However, it may not hold for a forum and blog post because in such a post the author may express opinions on multiple products and compare them using comparative and superlative sentences.

Most existing techniques for document-level sentiment classification are based on supervised learning, although there are also some unsupervised methods. We give an introduction to them below.

2.1.1 Classification Based on Supervised Learning

Sentiment classification can obviously be formulated as a supervised learning problem with two class labels (positive and negative). Training and testing data used in existing research are mostly product reviews, which is not surprising due to the above assumption. Since each review at a typical review site already has a reviewer-assigned rating (e.g., 1-5 stars), training and testing data are readily available. Typically, a review with 4-5 stars is considered a positive review (thumbs-up), and a review with 1-2 stars is considered a negative review (thumbs-down).

Sentiment classification is similar to but also different from classic topic-based text classification, which classifies documents into predefined topic classes, e.g., politics, sciences, sports, etc. In topic-based classification, topic related words are important. However, in sentiment classification, topic-related words are unimportant. Instead, sentiment or opinion words that indicate positive or negative opinions are important, e.g., *great*, *excellent*, *amazing*, *horrible*, *bad*, *worst*, etc.

Existing supervised learning methods can be readily applied to sentiment classification, e.g., naïve Bayesian, and support vector machines (SVM), etc. Pang et al. [73] took this approach to classify movie reviews into two classes, positive and negative. It was shown that using unigrams (a bag of individual words) as features in classification performed well with either naïve Bayesian or SVM. Neutral reviews were not used in this work, which made the problem easier. Note that features here are data attributes used in machine learning, not object features referred to in the previous section.

Subsequent research used many more kinds of features and techniques in learning. As most machine learning applications, the main task of sentiment classification is to engineer a suitable set of features. Some of the example features used in research and possibly in practice are listed below. For a more comprehensive survey of features used, please refer to [72].

Terms and their frequency: These features are individual words or word n-grams and their frequency counts. In some cases, word positions may also be considered. The TF-IDF weighting scheme from information retrieval may be applied too. These features are also commonly used in traditional topic-based text classification. They have been shown quite effective in sentiment classification as well.

Part of speech tags: It was found in many early researches that adjectives are important indicators of subjectivities and opinions. Thus, adjectives have been treated as special features.

Opinion words and phrases: *Opinion words* are words that are commonly used to express positive or negative sentiments. For example, *beautiful*, *wonderful*, *good*, and *amazing* are positive opinion words, and *bad*, *poor*, and *terrible* are negative opinion words. Although many opinion words are adjectives and adverbs, nouns (e.g., *rubbish*, *junk*, and *crap*) and verbs (e.g., *hate* and *like*) can also indicate opinions. Apart from individual words, there are also opinion phrases and idioms, e.g., *cost someone an arm and a leg*. Opinion words and phrases are instrumental to sentiment analysis for obvious reasons. We will discuss them further later in this section.

Syntactic dependency: Words dependency based features generated from parsing or dependency trees are also tried by several researchers.

Negation: Clearly negation words are important because their appearances often change the opinion orientation. For example, the sentence "*I don't like this camera*" is negative. However, negation words must be handled with care because not all occurrences of such words mean negation. For example, "not" in "*not only … but also*" does not change the orientation direction. We will discuss these issues again in Section 3.2.

Apart from classification or prediction of positive or negative sentiments, research has also been done on predicting the rating scores (e.g., 1-5 stars) of reviews [71]. In this case, the problem is formulated as a regression problem since the rating scores are ordinal. Another interesting research direction that has been investigated is the transfer learning or domain adaptation as it has been shown that sentiment classification is highly sensitive to the domain from which the training data are extracted. A classifier trained using opinionated texts from one domain often performs poorly when it is applied or tested on opinionated texts from another domain. The reason is that words and even language constructs used in different domains for expressing opinions can be substantially different. To make matters worse, the same word in one domain may mean positive, but in another domain may mean negative. For example, as observed in [95], the adjective *unpredictable* may have a negative orientation in a car review (e.g., "unpredictable steering"), but it could have a positive orientation in a movie review (e.g., "unpredictable plot"). Thus, domain adaptation is needed. Existing research has used labeled data from one domain and unlabeled data from the target domain and general opinion words as features for adaptation [3, 6, 105].

2.1.2 Classification Based on Unsupervised Learning

It is not hard to imagine that opinion words and phrases are the dominating indicators for sentiment classification. Thus, using unsupervised learning based on such words and phrases would be quite natural. The method in [95] is such a technique. It performs classification based on some fixed syntactic phrases that are likely to be used to express opinions. The algorithm consists of three steps:

Step 1: It extracts phrases containing adjectives or adverbs. The reason for doing this is that research has shown that adjectives and adverbs are good indicators of subjectivity and opinions. However, although an isolated adjective may indicate subjectivity, there may be an insufficient context to determine its opinion orientation. Therefore, the algorithm extracts two consecutive words, where one member of the pair is an adjective/adverb and the other is a context word. Two consecutive words are extracted if their POS tags conform to any of the patterns in Table 1. For example, the pattern in line 2 means that two consecutive words are extracted if the first word is an adverb and the second word is an adjective, but the third word (which is not extracted) cannot be a noun.

Table 1. Patterns of POS tags for extracting two-word phrases

	First word	Second word	Third word (Not Extracted)
1.	JJ	NN or NNS	anything
2.	RB, RBR, or RBS	JJ	not NN nor NNS
3.	JJ	JJ	not NN nor NNS
4.	NN or NNS	JJ	not NN nor NNS
5.	RB, RBR, or RBS	VB, VBD, VBN, or VBG	anything

Example 7: In the sentence, "*This camera produces beautiful pictures*", "beautiful pictures" will be extracted as it satisfies the first pattern.

Step 2: It estimates the orientation of the extracted phrases using the *pointwise mutual information* (PMI) measure given in Equation 1:

$$PMI(term_1, term_2) = \log_2\left(\frac{\Pr(term_1 \wedge term_2)}{\Pr(term_1)\Pr(term_2)}\right). \qquad (1)$$

Here, $\Pr(term_1 \wedge term_2)$ is the co-occurrence probability of $term_1$ and $term_2$, and $\Pr(term_1)\Pr(term_2)$ gives the probability that the two terms co-occur if they are statistically independent. The ratio between $\Pr(term_1 \wedge term_2)$ and $\Pr(term_1)\Pr(term_2)$ is thus a measure of the degree of statistical dependence between them. The log of this ratio is the amount of information that we acquire about the presence of one of the words when we observe the other.

The opinion orientation (*oo*) of a phrase is computed based on its association with the positive reference word "excellent" and its association with the negative reference word "poor":

$$oo(phrase) = PMI(phrase, \text{“}excellent\text{”}) - PMI(phrase, \text{“}poor\text{”}). \qquad (2)$$

The probabilities are calculated by issuing queries to a search engine and collecting the number of *hits*. For each search query, a search engine usually gives the number of relevant documents to the query, which is the number of hits. Thus, by searching the two terms together and separately, we can estimate the probabilities in Equation 1. Turney [95] used the AltaVista search engine because it has a NEAR operator, which constrains the search to documents that contain the words within ten words of one another, in either order. Let *hits*(*query*) be the number of hits returned. Equation 2 can be rewritten as:

$$oo(phrase) = \log_2\left(\frac{hits(\text{phrase } NEAR \text{ "excellent"})\, hits(\text{"poor"})}{hits(\text{phrase } NEAR \text{ "poor"})\, hits(\text{"excellent"})}\right). \qquad (3)$$

Step 3: Given a review, the algorithm computes the average *oo* of all phrases in the review, and classifies the review as recommended if the average *oo* is positive, not recommended otherwise.

Apart from this method many other unsupervised methods exist. See [16] for another example.

2.2 Sentence-Level Subjectivity and Sentiment Classification

We now move to the sentence-level to perform the similar task [35, 79, 80, 98, 103, 104, 107].

Task: Given a sentence *s*, two sub-tasks are performed:

1. *Subjectivity classification*: Determine whether *s* is a subjective sentence or an objective sentence,
2. *Sentence-level sentiment classification*: If *s* is subjective, determine whether it expresses a positive or negative opinion.

Notice that the quintuple (*o*, *f*, *oo*, *h*, *t*) is not used in defining the task here because sentence-level classification is often an intermediate step. In most applications, one needs to know what object or features of the object the opinions are on. However, the two sub-tasks of the sentence-level classification are still very important because (1) it filters out those sentences which contain no opinion, and (2) after we know what objects and features of the objects are talked about in a sentence, this step helps to determine whether the opinions on the objects and their features are positive or negative.

Most existing researches study both problems, although some of them only focus on one. Both problems are classification problems. Thus, traditional supervised learning methods are again applicable. For example, one of the early works reported by Wiebe et al. [98] performed subjectivity classification using the naïve Bayesian classifier. Subsequent research also used other learning algorithms [35, 80, 103, 107].

One of the bottlenecks in applying supervised learning is the manual effort involved in annotating a large number of training examples. To save the manual labeling effort, a bootstrapping approach to label training data automatically is reported in [80, 81]. The algorithm works by first using two high precision classifiers (HP-Subj and HP-Obj) to automatically identify some subjective and objective sentences. The high-precision classifiers use lists of lexical items (single words or n-grams) that are good subjectivity clues. HP-Subj classifies a sentence as subjective if it contains two or more strong subjective clues. HP-Obj classifies a sentence as objective if there are no strongly subjective clues. These classifiers will give very high precision but low recall. The extracted sentences are then added to the training data to learn patterns. The patterns (which form the subjectivity classifiers in the next iteration) are then used to automatically identify more subjective and objective sentences, which are then added to the training set, and the next iteration of the algorithm begins.

For pattern learning, a set of syntactic templates are provided to restrict the kinds of patterns to be learned. Some example syntactic templates and example patterns are shown below.

Syntactic template	Example pattern
<subj> passive-verb	<subj> was satisfied
<subj> active-verb	<subj> complained
active-verb <dobj>	endorsed <dobj>
noun aux <dobj>	fact is <dobj>
passive-verb prep <np>	was worried about <np>

Before discussing algorithms which also perform sentiment classification of subjective sentences, let us point out an assumption made in much of the research on the topic.

Assumption of sentence-level sentiment classification: The sentence expresses a single opinion from a single opinion holder.

This assumption is only appropriate for simple sentences with a single opinion, e.g., "*The picture quality of this camera is amazing*." However, for compound sentences, a single sentence may express more than one opinion. For example, the sentence, "*The picture quality of this camera is amazing and so is the battery life, but the viewfinder is too small for such a great camera*", expresses both positive and negative opinions (one may say that it has a mixed opinion). For "picture quality" and "battery life", the sentence is positive, but for "viewfinder", it is negative. It is also positive for the camera as a whole.

In [107], Yu and Hazivassiloglou reported a study which tries to classify subjective sentences and also determine their opinion orientations. For subjective or opinion sentence identification, it applied supervised learning. Three learning methods were evaluated: sentence similarity, naïve Bayesian classification, and multiple naïve Bayesian classifiers. For sentiment classification of each identified subjective sentence, it used a similar method to the method in [95], but with many more seed words (rather than only two used in [95]), and the score function was log-likelihood ratio. The same problem is studied in [35] considering gradable adjectives. In [28], a semi-supervised learning method is applied, and in [46], the decision is made by simply summing up opinion words in a sentence. [47, 48, 49] build models to identify some specific types of opinions in reviews.

As we mentioned earlier, sentence-level classification is not suitable for compound sentences. Wilson et al. [103] pointed out that not only a single sentence may contain multiple opinions, but also both subjective and factual clauses. It is useful to pinpoint such clauses. It is also important to identify the strength of opinions. A study of automatic sentiment classification was presented to classify clauses of every sentence by the *strength* of the opinions being expressed in individual clauses, down to four levels deep (*neutral*, *low*, *medium*, and *high*). The strength of *neutral* indicates the absence of opinion or subjectivity. Strength classification thus subsumes the task of classifying language as subjective versus objective. In [104], the problem is studied further using supervised learning by considering contextual sentiment influencers such as negation (e.g., *not* and *never*) and contrary (e.g., *but* and *however*). A list of influencers can be found in [76].

Finally, as mentioned in Section 1, we should bear in mind that subjective sentences are only a subset of opinionated sentences, and many objective sentences can also imply opinions. Thus, to mine opinions from text one needs to mine them from both types of sentences.

2.3 Opinion Lexicon Generation

In preceding sections, we mentioned that opinion words are employed in many sentiment classification tasks. We now discuss how such words are generated. In the research literature, opinion words are also known as *polar words*, *opinion-bearing words*, and *sentiment words*. Positive opinion words are used to express desired states while negative opinion words are used to express undesired states. Examples of positive opinion words are: *beautiful*, *wonderful*, *good*, and *amazing*. Examples of negative opinion words are *bad*, *poor*, and *terrible*. Apart from individual words, there are also opinion phrases and idioms, e.g., *cost someone an arm and a leg*. Collectively, they are called the *opinion lexicon*. They are instrumental for sentiment analysis for obvious reasons.

Opinion words can, in fact, be divided into two types, the *base type* and the *comparative type*. All the examples above are of the base type. Opinion words of the comparative type are used to express comparative and superlative opinions. Examples of such words are *better*, *worse*, *best*, *worst*, etc, which are comparative and superlative forms of their base adjectives or adverbs, e.g., *good* and *bad*. Unlike opinion words of the base type, the words of the comparative type do not express a direction opinion/sentiment on an object, but a comparative opinion/sentiment on more than one object, e.g., "*Car-x is better than Car-y*". This sentence tells something quite interesting. It does not express an opinion that any of the two cars is good or bad. It just says that comparing to Car-y, Car-x is better, and comparing to Car-x, Car-y is worse. Thus, although we still can assign a comparative word as positive or negative based on whether it represents a desirable or undesirable state, we cannot use it in the same way as an opinion word of the base type. We will discuss this issue further when we study sentiment analysis of comparative sentences. This section focuses on opinion words of the base type.

To compile or collect the opinion word list, three main approaches have been investigated: manual approach, dictionary-based approach, and corpus-based approach. Manual approach is very time-consuming [15, 65, 94, 106] and thus it is not usually used alone, but combined with automated approaches as the final check because automated methods make mistakes. Below, we discuss the two automated approaches.

Dictionary based approach: One of the simple techniques in this approach is based on bootstrapping using a small set of seed opinion words and an online dictionary, e.g., WordNet [25]. The strategy is to first collect a small set of opinion words manually with known orientations, and then to grow this set by searching in the WordNet for their synonyms and antonyms. The newly found words are added to the seed list. The next iteration starts. The iterative process stops when no more new words are found. This approach is used in [36, 46]. After the process completes, manual inspection can be carried out to remove and/or correct errors. Researchers have also used additional information (e.g., glosses) in WordNet and additional techniques (e.g., machine learning) to generate better lists [1, 21, 22, 24, 43]. So far, several opinion word lists have been generated [19, 23, 36, 87, 98].

The dictionary based approach and the opinion words collected from it have a major shortcoming. The approach is unable to find opinion words with domain specific orientations, which is quite common. For example, for a speakerphone, if it is quiet, it is usually negative. However, for a car, if it is quiet, it is positive. The corpus-based approach can help deal with this problem.

Corpus-based approach and sentiment consistency: The methods in the corpus-based approach rely on syntactic or co-occurrence patterns and also a seed list of opinion words to find other opinion words in a large corpus. One of the key ideas is the one proposed by Hazivassiloglou and McKeown [34]. The technique starts with a list of seed opinion adjective words, and uses them and a set of linguistic constraints or conventions on connectives to identify additional adjective opinion words and their orientations. One of the constraints is about conjunction (AND), which says that conjoined adjectives usually have the same orientation. For example, in the sentence, "*This car is beautiful **and** spacious*," if "beautiful" is known to be positive, it can be inferred that "spacious" is also positive. This is so because people usually express the same opinion on both sides of a conjunction. The following sentence is rather unnatural, "*This car is beautiful and difficult to drive*". If it is changed to "*This car is beautiful but difficult to drive*", it becomes acceptable. Rules or constraints are also designed for other connectives, OR, BUT, EITHER-OR, and NEITHER-NOR. We call this idea *sentiment consistency*. Of course, in practice it is not always consistent. Learning using the log-linear model is applied to a large corpus to determine if two conjoined adjectives are of the same or different orientations. Same and different-orientation links between adjectives forms a graph. Finally, clustering is performed on the graph to produce two sets of words: positive and negative. In [44], Kanayama and Nasukawa expanded this approach by introducing the idea of intra-sentential (within a sentence) and inter-sentential (between neighboring sentences) sentiment consistency (called *coherency* in [44]). The intra-sentential consistency is similar to that in [34]. Inter-sentential consistency applies the idea to neighboring sentences. That is, the same opinion orientation (positive or negative) is usually expressed in a few consecutive sentences. Opinion changes are indicated by adversative expressions such as *but* and *however*. Some criteria to determine whether to add a word to the positive or negative lexicon are also proposed. This study was based on Japanese text. Other related work include [42, 100].

In [78], Qiu et al proposed another method to extract domain specific sentiment words from reviews using also some seed opinion words. The main idea is to exploit certain syntactic relations of opinion words and object features for extraction. They showed that opinion words are almost always associated with object features in some ways. Thus, opinion words can be recognized by identified features, and features can be identified by known opinion words (no seed feature is needed). The extracted opinion words and features are utilized to identify new opinion words and new features, which are used again to extract more opinion words and features. This propagation or bootstrapping process ends when no more opinion words or features can be found. As the process involves propagation through both opinion words and features, the method is called *double propagation*. The extraction rules are designed based on different relations between opinion words and features, and also opinion words and features themselves. Dependency grammar [91] was adopted to describe these relations.

Using the corpus-based approach alone to identify all opinion words, however, is not as effective as the dictionary-based approach because it is hard to prepare a huge corpus to cover all English words. However, as we mentioned above, this approach has a major advantage that the dictionary-based approach does not have. It can help find domain specific opinion words and their orientations if a corpus from only the specific domain is used in the discovery process.

In [19], Ding and Liu explores the idea of intra-sentential and inter-sentential sentiment consistency further. Instead of finding domain dependent opinion words, they showed that the same word might have different orientations in different contexts even in the same domain. For example, in the digital camera domain, the word *long* expresses different opinions in the two sentences: "*The battery life is **long***" (positive) and "*The time taken to focus is **long***" (negative). Thus, finding domain dependent opinion words is still insufficient. They then proposed to consider both opinion words and object features together, and use the pair (*object_feature*, *opinion_word*) as the *opinion context*. Their method thus determines opinion words and their orientations together with the object features that they modify. The above rules about connectives were still applied. The work in [29] adopts the same context definition but used it for sentiment analysis of comparative sentences. In fact, the method in [90, 95] can also be considered as a method for finding context specific opinions. However, it does not use the sentiment consistency idea. Its opinion context is based on syntactic POS patterns rather than object features and opinion words that modify them. In [8], Breck et al. went further to study the problem of extracting any opinion expressions, which can have any number of words. The Conditional Random Fields (CRF) method [54] was used as the sequence learning technique for extraction.

Finally, we should note that populating an opinion lexicon (domain dependent or not) is different from determining whether a word or phrase is actually expressing an opinion and what its orientation is in a particular sentence. Just because a word or phrase is listed in an opinion lexicon does not mean that it actually is expressing an opinion in a sentence. For example, in the sentence, "*I am looking for a good health insurance for my family*," "good" here does not express either a positive or negative opinion on any particular insurance. And the same is true for polarity/opinion orientation. We should also realize that opinion words and phrases are not the only expressions that bear opinions. There are many others as we will see in Section 3.2 when we discuss rules of opinions.

3. Feature-Based Sentiment Analysis

Although classifying opinionated texts at the document level or at the sentence level is useful in many cases, they do not provide the necessary detail needed for some other applications. A positive opinionated document on a particular object does not mean that the author has positive opinions on all aspects or features of the object. Likewise, a negative opinionated document does not mean that the author dislikes everything. In a typical opinionated text, the author writes both positive and negative aspects of the object, although the general sentiment on the object may be positive or negative. Document-level and sentence-level classification does not provide such information. To obtain such details, we need to go to the object feature level, which means we need the full model of Section 1. Recall, at the feature level, the mining task is to discover every quintuple (o_j, f_{jk}, oo_{ijkl}, h_i, t_l) and identify all the synonyms (W_{jk}) and feature indicators I_{jk} of feature f_{jk}. In this section, we mainly focus on two key mining tasks:

1. Identify *object features* that have been commented on. For instance, in the sentence, "*The picture quality of this camera is amazing*," the object feature is "picture quality".
2. Determine whether the opinions on the features are positive, negative or neutral. In the above sentence, the opinion on the feature "picture quality" is positive.

Opinion holder, object and time extraction: In some applications, it is useful to identify and extract opinion holders, i.e., persons or organizations that expressed certain opinions. As we mentioned earlier, opinion holders are more useful for news articles or other types of formal documents, in which the persons or organizations who expressed opinions are stated explicitly in the text. Such holders need to be identified by the system [5, 14, 46]. In the case of the user-generated content on the Web, the opinion holders are often the authors of discussion posts, bloggers, or reviewers, whose login ids are known although their true identities in the real world may be unknown.

However, object name extraction is needed for discussion posts, blogs and also reviews. Note that although a review focuses on evaluating a particular object, it may compare it with other competing objects. Time extraction is also needed in the Web context. Since each web site usually displays the time when every post is submitted. So, the extraction is easy. However, in news and other types of documents time extraction is also an issue. All these three extraction tasks are collectively known as the Named Entity Recognition (NER) in the information extraction community. They have been studied extensively. See a comprehensive survey of information extraction tasks and algorithms in [82]. There is also a chapter in this book on information extraction.

Coreference resolution: In product reviews, the reviewed objects are usually known. However, this is not the case for opinions expressed in blogs and discussion posts. For example, in the post, "*I have a Canon S50 camera purchased from Amazon. It takes great photos*," two interesting questions can be asked: (1) what object does the post praise and (2) what "it" means in the second sentence? Clearly, we humans know that the post praises "Canon S50 camera", which is the problem of object extraction discussed above, and we also know that "it" here means "Canon S50 camera", which is the problem of coreference resolution. Coreference resolution has been studied extensively in NLP. However, it is still a major challenge. We will not discuss it here. A study of the problem in the sentiment analysis context is reported in [88].

In the next two subsections, we focus on the two tasks listed above.

3.1 Feature Extraction

Current research on object feature extraction is mainly carried out in online product reviews. We thus also focus on such reviews here. There are two common review formats on the Web. Different formats may need different techniques to perform the feature extraction task [55, 56].

Format 1 – Pros, cons and the detailed review: The reviewer is asked to describe Pros and Cons separately and also write a detailed/full review. An example of such a review is given in Figure 3.

Format 2 – Free format: The reviewer can write freely, i.e., no separation of Pros and Cons. An example of such a review is given in Figure 4.

3.1.1 Feature Extraction from Pros and Cons of Format 1

We describe a supervised pattern learning approach to extracting product features from Pros and Cons in reviews of Format 1 (not the detailed review, which is the same as that in Format 2). The key observation is that Pros and Cons are usually very brief, consisting of short phrases or sentence segments. Each sentence segment contains only one feature, and sentence segments are separated by commas, periods, semi-colons, hyphens, &, *and*, *but*, etc.

Example 8: Pros in Figure 3 can be separated into three segments:

great photos	⟨photo⟩
easy to use	⟨use⟩
very small	⟨small⟩ ⇒ ⟨size⟩.

Cons in Figure 3 can be separated into two segments:

battery usage	⟨battery⟩
included memory is stingy	⟨memory⟩

My SLR is on the shelf

by camerafun4. Aug 09 '04

Pros: Great photos, easy to use, very small
Cons: Battery usage; included memory is stingy.

I had never used a digital camera prior to purchasing this Canon A70. I have always used a SLR … **Read the full review**

Figure 3. An example review of Format 1

GREAT Camera., Jun 3, 2004

Reviewer: **jprice174** from Atlanta, Ga.

I did a lot of research last year before I bought this camera... It kinda hurt to leave behind my beloved nikon 35mm SLR, but I was going to Italy, and I needed something smaller, and digital.

The pictures coming out of this camera are amazing. The 'auto' feature takes great pictures most of the time. And with digital, you're not wasting film if the picture doesn't come out. …

Figure 4. An example review of Format 2

We can see that each segment describes a product feature, which is given within ⟨ ⟩. Notice that ⟨small⟩ is a feature indicator for feature ⟨size⟩. Clearly, there are many methods that can be used to extract features, e.g., Conditional Random Fields (CRF) [54]. Here, we describe a sequential rule based method [56].

The rules are called *label sequential rules* (LSR), which are generated from sequential patterns in data mining. A *label sequential rule* (LSR) is of the following form, $X \rightarrow Y$, where Y is a sequence and X is a sequence produced from Y by replacing some of its items with wildcards. A wildcard, denoted by a '*', can match any item.

The learning process is as follows: Each segment is first converted to a sequence. Each sequence element is a word, which is represented by both the word itself and its POS tag in a set. In the training data, all object features are manually labeled and replaced by the label $feature. An object feature can be expressed with a noun, adjective, verb or adverb. Thus, they represent both explicit features and implicit feature indicators. The labels and their POS tags used in mining LSRs are: {$feature, NN}, {$feature, JJ}, {$feature, VB} and {$feature, RB}, where $feature denotes a feature to be extracted, and NN stands for noun, VB for verb, JJ for adjective, and RB for adverb. Note that to simplify the presentation, we use NN and VB to represent all forms of nouns and verbs respectively.

For example, the sentence segment, "*Included memory is stingy*", is turned into the sequence

⟨{included, VB}{memory, NN}{is, VB}{stingy, JJ}⟩.

After labeling, it becomes (note that "memory" is an object feature):

⟨{included, VB}{$feature, NN}{is, VB}{stingy, JJ}⟩,

All the resulting sequences are then used to mine LSRs. An example rule is:

⟨{easy, JJ }{to}{*, VB}⟩ → ⟨{easy, JJ }{to}{$feature, VB}⟩ confidence = 90%

where the *confidence* is the conditional probability, $Pr(Y \mid X)$, which measures the accuracy of the rule.

Feature extraction is performed by matching the patterns with each sentence segment in a new review to extract object features. That is, the word in the sentence segment that matches $feature in a pattern is extracted. In the pattern match, only the right-hand side of each rule is used. In rule generation, both the right- and the left-hand sides are needed to compute the conditional probability or confidence. Details of sequential pattern mining and LSR mining can be found in [55].

3.1.2 Feature Extraction from Reviews of Format 2

Pros and Cons of Format 1 mainly consist of short phrases and incomplete sentences. The reviews of Format 2 usually use complete sentences. To extract features from such reviews, the above algorithm can also be applied. However, experiments show that it is not effective because complete sentences are more complex and contain a large amount of noise. Below, we describe some unsupervised methods for finding explicit features that are nouns and noun phrases. The first method is due to [36]. The method requires a large number of reviews, and consists of two steps:

1. Finding frequent nouns and noun phrases. Nouns and noun phrases (or groups) are identified by using a POS tagger. Their occurrence frequencies are counted, and only the frequent ones are kept. A frequency threshold can be decided experimentally. The reason for using this approach is that when people comment on product features, the vocabulary that they use usually converges, and most product features are nouns. Thus, those nouns that are frequently talked about are usually genuine and important features. Irrelevant contents in reviews are often diverse and thus infrequent, i.e., they are quite different in different reviews. Thus, those nouns that are infrequent are likely to be non-features or less important features.
2. Finding infrequent features by making use of opinion words. Opinion words are usually adjectives and adverbs that express positive or negative opinions. The idea is as follows: The same opinion word can be used to describe different object features. Opinion words that modify frequent features can also modify infrequent features, and thus can be used to extract infrequent features. For example, "picture" is found to be a frequent feature, and we have the sentence,

   "The pictures are absolutely amazing."

   If we know that "amazing" is a positive opinion word, then "software" can also be extracted as a feature from the following sentence,

   "The software is amazing."

   because the two sentences follow the same pattern and "software" in the sentence is also a noun.

The precision of step 1 of the above algorithm was improved by Popescu and Etzioni in [77]. Their algorithm tries to remove those noun phrases that may not be product features. It evaluates each noun phrase by computing a pointwise mutual information (PMI) score between the phrase and *meronymy discriminators* associated with the product class, e.g., a scanner class. The meronymy discriminators for the scanner class are, "of scanner", "scanner has", "scanner comes with", etc., which are used to find components or parts of scanners by searching on the Web. The PMI measure is a simplified version of the measure in [95] (also see Section 2.1.2).

$$PMI(f,d) = \frac{hits(f \wedge d)}{hits(f)hits(d)}, \tag{4}$$

where f is a candidate feature identified in step 1 and d is a discriminator. Web search is used to find the number of hits of individuals and also their co-occurrences. The idea of this approach is clear. If the PMI value of a candidate feature is too low, it may not be a component of the product because f and d do not co-occur frequently. The algorithm also distinguishes components/parts from attributes/properties using WordNet's *is-a* hierarchy (which enumerates different kinds of properties) and morphological cues (e.g., "-iness", "-ity" suffixes).

The double propagation method in [78], which has been described in Section 2.3, can also be used to extract features. It in fact exploits and extends the idea in step 2 above (without using step 1), and starts with only a set of seed opinion words (no seed features required). That is, it utilizes the association or dependency relations of opinion words and features, i.e., opinion words always modify features. The associations are described using the dependency grammar [91], which results in a set of syntactic rules for the extraction of both opinion words and object features in an iterative fashion.

Other related works on feature extraction mainly use the ideas of topic modeling and clustering to capture topics/features in reviews [58, 62, 89, 93, 106]. For example, in [63], Mei et al. proposed a probabilistic model called *topic-sentiment mixture* to capture the mixture of features and sentiments simultaneously. One topic model and two sentiment models were defined based on language models to capture the probabilistic distribution of words in different topics/features with their associated opinion orientations. Su et al. [89] also proposed a clustering based method with mutual reinforcement to identify implicit features.

After the extraction of object features, two additional problems need to be solved:

Group synonyms: It is common that people use different words or phrases to describe the same feature. For example, *photo* and *picture* refer to the same feature in digital camera reviews. Identifying and grouping synonyms is essential for applications. Although WordNet [25] and other thesaurus dictionaries help to some extent, they are far from sufficient due to the fact that many synonyms are domain dependent. For example, *picture* and *movie* are synonyms in movie reviews, but they are not synonyms in digital camera reviews as *picture* is more related to *photo* while *movie* refers to *video*. Carenini et al. [10] proposed a method based on several similarity metrics similar to those in information integration [55]. It requires a taxonomy of features to be given for a particular domain. The algorithm merges each discovered feature to a feature node in the taxonomy. The similarity metrics are defined based on string similarity, synonyms and other distances measured using WordNet. Experiments based on digital camera and DVD reviews show promising results.

Mapping to implicit features: Feature extraction may discover many feature indicators. Adjectives and adverbs are perhaps the most common type of feature indicators. It is known that many adjectives and adverbs modify or describe some specific attributes or properties of objects. This step maps such feature indicators to features. For example, the adjective *heavy* usually describes the *weight* of an object, and thus should be mapped to the *weight* feature. *Beautiful* is usually used to describe the *appearance* of an object, and thus should be mapped to the *appearance* feature. However, this needs to be done with care as the usage of many adjectives can be quite versatile. Their exact meaning may be domain/context dependent. For example, "heavy" in the sentence "*The traffic is heavy*" does not describe the *weight* of the traffic. One way to map indicator words to (implicit) features is to manually compile a list of such mappings during training data annotation, which can then be used in the same domain in the future. However, it is not clear whether this is an effective approach as little research has been done.

3.2. Opinion Orientation Identification

We now discuss how to identify the orientation of opinions expressed on an object feature in a sentence. Clearly, the sentence-level and clause-level sentiment classification methods discussed in Section 2 are applicable here. That is, they can be applied to each sentence or clause which contains object features, and the features in it will take its opinion orientation. Here, we only describe a *lexicon-based approach* to solving the problem [19, 36]. See a more complex method based on relaxation labeling in [77].

The lexicon-based approach basically uses *opinion words* and *phrases* in a sentence to determine the orientation of the opinion. Apart from the opinion lexicon, negations and *but*-clauses in a sentence are also crucial and need to be handled. The approach works as follows [36, 19]:

1. **Identifying opinion words and phrases**: Given a sentence that contains an object feature, this step identifies all opinion words and phrases. Each positive word is assigned the opinion score of +1, each negative word is assigned the opinion score of -1, and each context dependent word is assigned the opinion score of 0. For example, we have the sentence, "*The picture quality of this camera is not great, but the battery life is long*." After this step, the sentence is turned into "The *picture quality* of this camera is not **great**[+1], but the *battery life* is **long**[0]" because "great" is a positive opinion word and "long" is context dependent. The object features are italicized.
2. **Handling negations**: Negation words and phrases are used to revise the opinion scores obtained in

step 1 based on some negation handling rules. After this step, the above sentence is turned into "The *picture quality* of this camera is not **great**[-1], but the *battery life* is **long**[0]" due to the negation word "not". We note that not every "not" means negation, e.g., "not only … but also". Such *non-negation phrases containing negation words* need to be considered separately.

3. **But-clauses**: In English, *but* means *contrary*. A sentence containing *but* is handled by applying the following rule: the opinion orientation before *but* and after *but* are opposite to each other. After this step, the above sentence is turned into "The *picture quality* of this camera is not **great**[-1], but the *battery life* is **long**[+1]" due to "but". Apart from *but*, phrases such as "*with the exception of*", "*except that*", and "*except for*" behave similarly to *but* and are handled in the same way. As in the case of negation, not every *but* means contrary, e.g., "*not only … but also*". Such *non-but phrases containing "but"* also need to be considered separately.
4. **Aggregating opinions**: This step applies an opinion aggregation function to the resulting opinion scores to determine the final orientation of the opinion on each object feature in the sentence. Let the sentence be s, which contains a set of object features $\{f_1, \ldots, f_m\}$ and a set of opinion words or phrases $\{op_1, \ldots, op_n\}$ with their opinion scores obtained from steps 1, 2 and 3. The opinion orientation on each feature f_i in s is determined by the following opinion aggregation function:

$$score(f_i, s) = \sum_{op_j \in s} \frac{op_j.so}{d(op_j, f_i)}, \quad (5)$$

where op_j is an opinion word in s, $d(op_j, f_i)$ is the distance between feature f_i and opinion word op_j in s. $op_j.so$ is the orientation or the opinion score of op_i. The multiplicative inverse in the formula is used to give low weights to opinion words that are far away from feature f_i. If the final score is positive, then the opinion on feature f_i in s is positive. If the final score is negative, then the opinion on the feature is negative. It is neutral otherwise.

This simple algorithm is useful but not sufficient in many cases. One major shortcoming is that opinion words and phrases do not cover all expressions that convey or imply opinions. There are in fact many others. Below, we present basic rules of opinions.

Basic Rules of Opinions

A rule of opinion is an implication with an expression on the left and an implied opinion on the right. The expression is a conceptual one as it represents a concept, which can be expressed in many ways in an actual sentence. The application of opinion words/phrases above can also be represented as such rules. Let Neg be a negative opinion word/phrase and Pos be a positive opinion word/phrase. The rules for applying opinion words/phrases in a sentence are as follow:

1. Neg → Negative
2. Pos → Positive

These rules say that Neg implies a negative opinion (denoted by *Negative*) and Pos implies a positive opinion (denoted by *Positive*) in a sentence. The effect of negations can be represented as well:

3. Negation Neg → Positive
4. Negation Pos → Negative

The rules state that negated opinion words/phrases take their opposite orientations in a sentence. Note that the above use of "*but*" is not considered an opinion rule but a language convention that people often use to indicate a possible opinion change. We now describe some additional rules of opinions.

Deviation from the norm or some desired value range: In some domains, an object feature may have an expected or desired value range or norm. If it is above and/or below the normal range, it is negative, e.g., "*This drug causes low* (or *high*) *blood pressure*." We then have the following rules

5. Desired value range → Positive

6. Below or above the desired value range → Negative

Decreased and increased quantities of opinionated items: This set of rules is similar to the negation rules above. Decreasing or increasing the quantities associated with some opinionated items may change the orientations of the opinions. For example, "*This drug reduced my pain significantly*." Here, "pain" is a negative opinion word, and the reduction of "pain" indicates a desirable effect of the drug. Hence, the decreased pain implies a positive opinion on the drug. The concept of "decreasing" also extends to "removal" or "disappearance", e.g., "*My pain has disappeared after taking the drug*."

7. Decreased Neg → Positive
8. Decreased Pos → Negative
9. Increased Neg → Negative
10. Increased Pos → Positive

The last two rules may not be needed as there is no change of opinion orientations.

Producing and consuming resources and wastes: If an object produces resources, it is positive. If it consumes resources, especially a large quantity of them, it is negative. For example, "money" is a resource. The sentence, "*Company-x charges a lot of money*" gives a negative opinion on "Company-x". Likewise, if an object produces wastes, it is negative. If it consumes wastes, it is positive. These give us the following rules:

11. Consume resource → Negative
12. Produce resource → Positive
13. Consume waste → Positive
14. Produce waste → Negative

These basic rules can also be combined to produce compound rules, e.g., "Consume decreased waste → Negative" which is a combination of rules 7 and 13. To build a practical system, all these rules and their combinations need to be considered.

As noted above, these are conceptual rules. They can be expressed in many ways using different words and phrases in an actual text, and in different domains they may also manifest differently. However, by no means, we claim these are the only basic rules that govern expressions of positive and negative opinions. With further research, additional new rules may be discovered and the current rules may be refined or revised. Neither do we claim that any manifestation of such rules imply opinions in a sentence. Like opinion words and phrases, just because a rule is satisfied in a sentence does not mean that it actually is expressing an opinion, which makes sentiment analysis a very challenging task.

4. Sentiment Analysis of Comparative Sentences

Directly expressing positive or negative opinions on an object and its features is only one form of evaluation. Comparing the object with some other similar objects is another. Comparisons are related to but are also quite different from direct opinions. They not only have different semantic meanings, but also different syntactic forms. For example, a typical direct opinion sentence is "*The picture quality of this camera is great*." A typical comparison sentence is "*The picture quality of Camera-x is better than that of Camera-y*." This section first defines the problem, and then presents some existing methods for their analysis [29, 38, 39].

4.1 Problem Definition

In general, a comparative sentence expresses a relation based on similarities or differences of more than one object. The comparison is usually conveyed using the *comparative* or *superlative* form of an adjective or adverb. A comparative is used to state that one object has more of a certain quantity than another object. A superlative is used to state that one object has the most or least of a certain quantity. In general, a comparison can be between two or more objects, groups of objects, and one object and the rest of the objects. It can also be between an object and its previous or future versions.

Two types of comparatives: In English, comparatives are usually formed by adding the suffix "*-er*" and superlatives are formed by adding the suffix "*–est*" to their *base adjectives* and *adverbs*. For example, in "*The battery life of Camera-x is longer than that of Camera-y*", "longer" is the comparative form of the adjective "*long*". In "*The battery life of this camera is the longest*", "longest" is the superlative form of the adjective "long". We call this type of comparatives and superlatives *Type 1 comparatives* and *superlatives*. For simplicity, we will use Type 1 comparatives to mean both from now on.

Adjectives and adverbs with two syllables or more and not ending in *y* do not form comparatives or superlatives by adding "*–er*" or "*–est*". Instead, *more*, *most*, *less* and *least* are used before such words, e.g., *more beautiful*. We call this type of comparatives and superlatives *Type 2 comparatives* and *Type 2 superlatives*. Both Type 1 and Type 2 are called *regular comparatives* and *superlatives* respectively.

In English, there are also some *irregular comparatives* and *superlatives*, which do not follow the above rules, i.e., *more*, *most*, *less*, *least*, *better*, *best*, *worse*, *worst*, *further/farther* and *furthest/farthest*. They behave similarly to Type 1 comparatives and superlatives and thus are grouped under Type 1.

Apart from these standard comparatives and superlatives, many other words can also be used to express comparisons, e.g., *prefer* and *superior*. For example, the sentence, "*Camera-x's quality is superior to Camera-y*", says that "Camera-x" is preferred. In [38], Jindal and Liu identified a list such words. Since these words behave similarly to Type 1 comparatives, they are also grouped under Type 1.

Further analysis also shows that comparatives can be grouped into two categories according to whether they express increased or decreased values, which are useful in sentiment analysis.

Increasing comparatives: Such a comparative expresses an increased value of a quantity, e.g., *more* and *longer*.

Decreasing comparatives: Such a comparative expresses a decreased value of a quantity, e.g., *less* and *fewer*.

Types of comparative relations: Comparative relations can be grouped into four main types. The first three types are called *gradable comparisons* and the last one is called the *non-gradable comparison*.

1. *Non-equal gradable comparisons*: Relations of the type *greater* or *less than* that express an ordering of some objects with regard to some of their features, e.g., "*The Intel chip is faster than that of AMD*". This type also includes user preferences, e.g., "*I prefer Intel to AMD*".
2. *Equative comparisons*: Relations of the type *equal to* that state two objects are equal with respect to some of their features, e.g., "*The picture quality of camera X is as good as that of camera Y*"
3. *Superlative comparisons*: Relations of the type *greater* or *less than all others* that rank one object over *all* others, e.g., "*The Intel chip is the fastest*".
4. *Non-gradable comparisons*: Relations that compare features of two or more objects, but do not grade them. There are three main sub-types:
   - Object *A* is similar to or different from object *B* with regard to some features, e.g., "*Coke tastes differently from Pepsi*"
   - Object *A* has feature f_1, and object *B* has feature f_2 (f_1 and f_2 are usually substitutable), e.g., "*desktop PCs use external speakers but laptops use internal speakers*"
   - Object *A* has feature *f*, but object *B* does not have, e.g., "*Cell phone X has an earphone, but cell phone Yoes not have*"

Mining objective: Given an opinionated document *d*, *comparison mining* consists of two tasks:

1. Identify comparative sentences in *d*, and classify the identified comparative sentences into different types or classes.
2. Extract comparative opinions from the identified sentences. A *comparative opinion* in a

comparative sentence is expressed with:

(O_1, O_2, F, po, h, t),

where O_1 and O_2 are the object sets being compared based on their shared features F (objects in O_1 appear before objects in O_2 in the sentence), po is the preferred object set of the opinion holder h, and t is the time when the comparative opinion is expressed.

As for direct opinions, not every piece of information is needed in an application. In many cases, h and t may not be required by applications.

Example 9: Consider the comparative sentence "*Canon's optics is better than those of Sony and Nikon.*" written by John on May 1, 2009. The extracted comparative opinion is:

({Canon}, {Sony, Nikon}, {optics}, *preferred*:{Canon}, John, May-1-2009).

The object set O_1 is {Canon}, the object set O_2 is {Sony, Nikon}, their shared feature set F being compared is {optics}, the preferred object set is {Canon}, the opinion holder h is John and the time t when this comparative opinion was written is May-1-2009.

Below, we study the problem of identifying comparative sentences and mining comparative opinions.

4.2 Identification of Comparative Sentences

Although most comparative sentences contain comparative adjectives and comparative adverbs, e.g., *better,* and *longer*, many sentences that contain such words are not comparatives, e.g., "*I cannot agree with you more*". Similarly, many sentences that do not contain such indicators are comparative sentences (usually non-gradable), e.g., "*Cellphone-x has Bluetooth, but Cellphone-y does not have*."

An interesting phenomenon about comparative sentences is that such a sentence usually has a keyword or a key phrase indicating comparison. It is shown in [38] that using a set of 83 keywords and key phrases, 98% of the comparative sentences (recall = 98%) can be identified with a precision of 32% using the authors' data set. The keywords and key phrases are:

1. Comparative adjectives (JJR) and comparative adverbs (RBR), e.g., *more*, *less*, *better*, and words ending with *-er*.
2. Superlative adjectives (JJS) and superlative adverbs (RBS), e.g., *most*, *least*, *best*, and words ending with *-est*.
3. Other indicative words such as *same*, *similar*, *differ*, *as well as*, *favor*, *beat*, *win, exceed*, *outperform, prefer*, *ahead, than*, *superior*, *inferior*, *number one, up against*, etc.

Since keywords alone are able to achieve a high recall, the set of keywords can be used to filter out those sentences that are unlikely to be comparative sentences. We can then improve the precision of the remaining set of sentences.

It is also observed that comparative sentences have strong patterns involving comparative keywords, which is not surprising. These patterns can be used as features in machine learning. To discover these patterns, class sequential rule (CSR) mining is used in [38]. Class sequential rule mining is a sequential pattern mining method from data mining. Each training example used for mining CSRs is a pair (s_i, y_i), where s_i is a sequence and y_i is a class, e.g., $y_i \in$ {*comparative*, *non-comparative*}. The sequence is generated from a sentence. Instead of using each full sentence, only words near a comparative keyword are used to generate each sequence. Each sequence is also labeled with a class indicating whether the sentence is a comparative sentence or not. Using the training data, CSRs can be generated. Details of the mining algorithm can be found in [38, 55].

For classification model building, the left-hand side sequence patterns of the rules with high conditional probabilities are used as data features in [38]. If the sentence matches a pattern, the corresponding feature value for the pattern is 1, and otherwise it is 0. Bayesian classification is employed for model building.

Classify comparative sentences into three types: This step classifies comparative sentences obtained from the last step into one of the three types, *non-equal gradable*, *equative*, and *superlative* (non-gradable may also be added). For this task, keywords alone are already sufficient. That is, the set of keywords is used as data features for machine learning. It is shown in [38] that SVM gives the best results.

4.3 Extraction of Objects and Object Features in Comparative Sentences

To extract objects and object features being compared, many information extraction methods can be applied, e.g., Conditional Random Fields (CRF), Hidden Markov Models (HMM), and others. For a survey of information extraction techniques, see [82]. In [39], Jindal and Liu used label sequential rules (LSR) and CRF to perform the extraction. The algorithm makes the following assumptions:

1. There is only one comparative relation in a sentence. In practice, this is violated only in a very small number of cases.
2. Objects or their features are nouns (includes nouns, plural nouns and proper nouns) and pronouns. These cover most cases. However, a feature can sometimes be a noun used in its verb form or some action described as a verb (e.g., "*Intel costs more*"; "costs" is a verb and an object feature). These are adverbial comparisons and are not considered in [39].

In [7], Bos and Nissim also proposed a method to extract some useful items from superlative sentences.

4.4 Identification of Preferred Objects in Comparative Sentences

Similar to sentiment analysis of normal sentences, sentiment analysis of comparative sentences also needs to determine whether a comparative sentence is opinionated or not. However, unlike normal sentences, it does not make good sense to apply sentiment classification to comparative sentences because an opinionated comparative sentence does not express a direct positive or negative opinion. Instead, it compares multiple objects by ranking the objects based on their shared features to give a *comparative opinion*. In other words, it presents a preference order of the objects based on the comparison of some of their shared features. Since most comparative sentences compare only two sets of objects, analysis of an opinionated comparative sentence means to identify the preferred object set. Since little research has been done on classifying whether a comparative sentence is opinionated or not, below we only briefly describe a method [29] for identifying the preferred objects.

The approach bears some resemblance to the lexicon-based approach to identifying opinion orientations on object features. Thus, it needs opinion words used for comparisons. Similar to normal opinion words, these words can also be divided into two categories.

1. *Comparative opinion words*: For Type 1 comparatives, this category includes words such as *better*, *worse*, etc, which have explicit and domain independent opinions. In sentences involving such words, it is normally easy to determine which object set is the preferred one of the sentence author.

   In the case of Type 2 comparatives, formed by adding *more*, *less*, *most*, and *least* before adjectives/adverbs, the preferred object set is determined by both words. The following rules apply:

   <Increasing Comparative> Negative → Negative Comparative Opinion
   <Increasing Comparative> Positive → Positive Comparative Opinion
   <Decreasing Comparative> Negative → Positive Comparative Opinion
   <Decreasing Comparative> Positive → Negative Comparative Opinion

   The first rule says that the combination of an increasing comparative (e.g., *more*) and a negative opinion word (e.g., *awful*) implies a negative Type 2 comparative. The other rules are similar. Note that the positive (or negative) opinion word is of the base type, while the positive (or negative) comparative opinion is of the comparative type.

2. *Context-dependent comparative opinion words*: In the case of Type 1 comparatives, such words include *higher*, *lower*, etc. For example, "*Car-x has higher mileage per gallon than Car-y*" carries a

positive sentiment on "Car-x" and a negative sentiment on "Car-y" comparatively, i.e., "Car-x" is preferred. However, without domain knowledge it is hard to know whether "higher" is positive or negative. The combination of "higher" and "mileage" with the domain knowledge tells us that "higher mileage" is desirable.

In the case of Type 2 comparatives, the situation is similar. However, in this case, the comparative word (*more*, *most*, *less* or *least*), the adjective/adverb and the object feature are all important in determining the opinion or preference. If we know whether the comparative word is increasing or decreasing (which is easy since there are only four of them), then the opinion can be determined by applying the four rules in (1) above.

As discussed in Section 2.3, the pair (*object_feature*, *opinion_word*) forms an opinion context. To determine whether a pair is positive or negative, the algorithm in [29] resorts to external information, i.e., a large corpus of Pros and Cons from product reviews. It basically determines whether the object_feature and opinion_word are more associated with each other in Pros or in Cons. If they are more associated in Pros, it is positive. Otherwise, it is negative. Using Pros and Cons is natural because they are short phrases and thus have little noise, and their opinion orientations are known.

To obtain comparative opinion words, due to the observation below we can simply convert opinion adjectives/adverbs to their comparative forms, which can be done automatically based on the English comparative formation rules described above and the WordNet.

Observation: If an adjective or adverb is positive (or negative), then its comparative or superlative form is also positive (or negative), e.g., *good*, *better* and *best*.

After the conversion, these words are manually categorized into increasing and decreasing comparatives.

Once all the information is available, determining which object set is preferred is relatively simple. Without negation, if the comparative is positive (or negative), then the objects before (or after) *than* is preferred. Otherwise, the objects after (or before) *than* are preferred. Additional details can be found in [29]. In [26], Fiszman et al. studied the problem of identifying which object has more of certain features in comparative sentences in biomedical texts, but it does not analyze opinions.

5. Opinion Search and Retrieval

As Web search has proven to be very important, it is not hard to imagine that opinion search will also be of great use. One can crawl the user-generated content on the Web and enable people to search for opinions on any subject matter. Two typical kinds of opinion search queries may be issued:

1. find public opinions on a particular object or a feature of the object, e.g., find customer opinions on a digital camera or the picture quality of the camera, and find public opinions on a political topic. Recall that an object can be a product, organization, event, or topic.
2. find opinions of a person or organization (i.e., opinion holder) on a particular object or a feature of the object, e.g., find Barack Obama's opinion on abortion. This type of search is particularly relevant to news articles, where individuals or organizations who express opinions are explicitly stated.

For the first type of queries, the user may simply give the name of the object or the name of the feature and the name of the object. For the second type of queries, the user may give the name of the opinion holder and the name of the object.

Similar to traditional Web search, opinion search also has two major tasks: 1) retrieving relevant documents/sentences to the user query, and 2) ranking the retrieved documents/sentences. However, there are also major differences. On retrieval, opinion search needs to perform two sub-tasks:

1. Find documents or sentences that are relevant to the query topic. This is the only task performed in the traditional Web search or information retrieval.
2. Determine whether the documents or sentences express opinions and whether the opinions are

positive or negative. This is the task of sentiment analysis. Traditional search does not perform this sub-task. It is this sub-task that makes the opinion search more complex than traditional search.

As for ranking, traditional Web search engines rank Web pages based on authority and relevance scores [55]. The basic premise is that the top ranked pages (ideally the first page) contain sufficient information to satisfy the user's information need. This paradigm is adequate for factual information search because *one fact equals to any number of the same fact*. That is, if the first page contains the required information, there is no need to see the rest of the relevant pages. For opinion search, this paradigm is fine for the second type of queries because the opinion holder usually has only one opinion on a particular object or topic, and the opinion is contained in a single document or page. However, for the first type of opinion queries, this paradigm needs to be modified because ranking in opinion search has two objectives. First, it needs to rank those opinionated documents or sentences with high utilities or information contents at the top (see Section 6.2). Second, it also needs to reflect the natural distribution of positive and negative opinions. This second objective is important because in most practical applications the actual proportions of positive and negative opinions are the most important pieces of information as in traditional opinion surveys. Only reading the top ranked results as in the traditional search is problematic because *one opinion does not equal to multiple opinions*. The top result only represents the opinion of a single person or organization. Thus, ranking in opinion search needs to capture the natural distribution of the positive and negative sentiments of the whole population. One simple solution is to produce two rankings, one for positive opinions and one for negative opinions. The numbers of positive and negative opinions indicate the distribution.

Providing a feature-based summary for each opinion search will be even better. However, it is an extremely challenging problem as we have seen that feature extraction, feature grouping and associating objects to its features are all very difficult problems. Like opinion search, comparison search will be useful too. For example, when one wants to register for a free email account, one most probably wants to know which email system is the best for him/her, e.g., hotmail, gmail or *Yahoo*! mail. Wouldn't it be nice if one can find comparisons of features of these email systems from existing users by issuing a search query "hotmail vs. gmail vs. yahoo mail."? So far, little research has been done in this direction although the work in [29, 38, 39] can be of use in this context.

To give a favor of what an opinion search system looks like, we present an example system [109], which is the winner of the blog track in the 2007 TREC evaluation (http://trec.nist.gov/). The task of this track is exactly opinion search (or retrieval). This system has two components. The first component is for retrieving relevant documents for each query. The second component is for classifying the retrieved documents as opinionated or not-opinionated (subjectivity classification). The opinionated documents are further classified into positive, negative or mixed (containing both positive and negative opinions).

Retrieval component: This component performs the traditional information retrieval (IR) task. Unlike a normal IR system, which is based on keyword match, this component considers both keywords and concepts. Concepts are named entities (e.g., names of people or organizations) or various types of phrases from dictionaries and other sources (e.g., Wikipedia entries). The strategy for processing a user query is as follows [108, 109]: It first recognizes and disambiguates the concepts within the user query. It then broadens the search query with its synonyms. After that, it recognizes concepts in the retrieved documents, and also performs pseudo-feedback to automatically extract relevant words from the top-ranked documents to expand the query. Finally, it computes a similarity (or relevance score) of each document with the expanded query using both concepts and keywords.

Opinion classification component: This component performs two tasks: (1) classifying each document into one of the two categories, opinionated and not-opinionated, and (2) classifying each opinionated document as expressing a positive, negative or mixed opinion. For both tasks, the system uses supervised learning. For the first task, it obtains a large amount of opinionated (subjective) training data from review sites such as rateitall.com and epinion.com. The data are also collected from different domains involving consumer goods and services as well as government policies and political viewpoints. The not-opinionated training data are obtained from sites that give objective information such as Wikipedia. From these training data, a SVM classifier is constructed.

This classifier is then applied to each retrieved document as follows: The document is first partitioned into sentences. The SVM classifier then classifies a sentence as opinionated or not-opinionated. If a sentence is classified to be opinionated, its strength as determined by SVM is also noted. A document is regarded opinionated if there is at least one sentence that is classified as opinionated. To ensure that the opinion of the sentence is directed to the query topic, the system requires that enough query concepts/words are found in its vicinity. The totality of the opinionated sentences and their strengths in a document together with the document's similarity with the query is used to rank the document relative to other documents.

To determine whether an opinionated document express a positive, negative or mixed opinion, the second classifier is constructed. The training data are reviews from review sites containing review ratings (e.g., rateitall.com). A low rating indicates a negative opinion while a high rating indicates a positive opinion. Using positive and negative reviews as training data, a sentiment classifier is built to classify each document as expressing positive, negative, or mixed opinion.

There are many other approaches for opinion retrieval. The readers are encouraged to read the papers at the TREC site (http://trec.nist.gov/pubs/trec16/t16_proceedings.html), and the overview paper of 2007 TREC blog track [60]. Other related work includes [20, 27, 66].

6. Opinion Spam and Utility of Opinions

Email spam and Web spam are quite familiar to most people. Email spam refers to unsolicited commercial emails selling products and services, while Web spam refers to the use of "illegitimate means" to boost the search rank positions of target Web pages. The reason for spam is mainly due to economics. For example, in the Web context, the economic and/or publicity value of the rank position of a page returned by a search engine is of great importance. If someone searches for a product that your Web site sells, but the product page of your site is ranked very low (e.g., beyond the top 20) by a search engine, then the chance that the person will go to your page is extremely low, let alone to buy the product from your site. This is certainly bad for the business. There are now many companies that are in the business of helping others improve their page ranking by exploiting the characteristics and weaknesses of current search ranking algorithms. These companies are called *Search Engine Optimization* (SEO) companies. Some SEO activities are ethical and some, which generate spam, are not. For more information on Web spam, please refer to [55].

In the context of opinions, we have a similar spam problem [40, 41]. Due to the explosive growth of the user-generated content, it has become a common practice for people to find and to read opinions on the Web for many purposes. For example, a person plans to buy a camera. Most probably, he/she will go to a merchant or review site (e.g., amazon.com) to read the reviews of some cameras. If he/she find that most reviews are positive about a camera, he/she is very likely to buy the camera. However, if most reviews are negative, he/she will almost certainly choose another camera. Positive opinions can result in significant financial gains and/or fames for organizations and individuals. This, unfortunately, also gives good incentives for *opinion spam*, which refers to human activities (e.g., write spam reviews) that try to deliberately mislead readers or automated opinion mining systems by giving undeserving positive opinions to some target objects in order to promote the objects and/or by giving unjust or false negative opinions to some other objects to damage their reputations. Such opinions are also called *fake opinions* or *bogus opinions*. They have become an intense discussion topic in blogs and forums, and also in press (e.g., http://travel.nytimes.com/2006/02/07/business/07guides.html), which show that review spam has become a problem. We can predict that as opinions on the Web are increasingly used in practice by consumers and organizations, the problem of detecting spam opinions will become more and more critical.

A related problem that has also been studied in the past few years is the determination of the usefulness, helpfulness or utility of a review [31, 49, 57, 110]. The idea is to determine how helpful a review is to a user. This is a useful task as it is desirable to rank reviews based on utilities or qualities when showing the reviews to the user, with the most useful reviews at the top. In fact, many review aggregation sites have been practicing this for years. They obtain the helpfulness or utility score of each review by asking readers to provide helpfulness feedbacks to each review. For example, in amazon.com, the reader can indicate whether he/she finds a review helpful by responding to the question "*Was the review helpful to you?*" just below each review. The feedback results from all those responded are then aggregated and displayed right before each review, e.g., "*15 of 16 people found the following review helpful*". Although most review sites already provide the service, automatically determining the quality or the usefulness of a review is still useful because many reviews have few or no feedbacks. This is especially true for new reviews and reviews of products that are not very popular.

This section uses customer reviews of products as an example to study opinion spam and utilities of opinions. However, most of the analyses are also applicable to opinions expressed in other forms of the user-generated content, e.g., forum posts and blogs.

6.1 Opinion Spam

There are generally three types of spam reviews as defined by Jindal and Liu in [40, 41].

- **Type 1 (untruthful opinions)**: These are reviews that deliberately mislead readers or opinion mining systems by giving undeserving positive reviews to some target objects in order to promote the objects and/or by giving unjust or malicious negative reviews to some other objects in order to damage their reputation. Untruthful reviews are also commonly known as fake reviews or bogus reviews as we mentioned earlier.
- **Type 2 (opinions on brands only)**: These are reviews that do not comment on the specific products that they are supposed to review, but only comment on the brands, the manufacturers or the sellers of the products. Although they may be useful, they are considered as spam because they are not targeted at the specific products and are often biased. For example, in a review for a HP printer, the reviewer only wrote "*I hate HP. I never buy any of their products*".
- **Type 3 (non-opinions)**: These are not reviews or opinionated although they appear as reviews. There are two main sub-types: (1) advertisements, and (2) other irrelevant texts containing no opinions (e.g., questions, answers, and random texts).

In general, spam detection can be formulated as a classification problem with two classes, *spam* and *non-spam*. Due to the specific nature of the different types of spam, they need to be dealt with differently. For spam reviews of type 2 and type 3, they can be detected based on traditional classification learning using manually labeled spam and non-spam reviews because these two types of spam reviews are easily recognizable manually. The main task is to find a set of effective data features for model building. Note again that here the features refer to features in machine learning not object features used in feature-based sentiment analysis. In [40, 41], three sets of features were identified for learning,

Review centric features: These are features about the content of each review. Example features are actual text of the review, number of times that brand names are mentioned, percentage of opinion words, review length, and number of helpful feedbacks.

Reviewer centric features: These are features about a reviewer. Example features are average rating given by the reviewer, standard deviation in rating, ratio of the number of reviews that the reviewer wrote which were the first reviews of the products to the total number of reviews that he/she wrote, and ratio of the number of cases in which he/she was the only reviewer.

Product centric features: These are features about each product. Example features are price of the product, sales rank of the product (amazon.com assigns sales rank to "now selling products" according to their sales volumes), average rating, and standard deviation in ratings of the reviews on the product.

Table 2. Spam reviews vs. product quality

	Positive spam review	Negative spam review
Good quality product	1	***2***
Bad quality product	***3***	4
Average quality product	***5***	***6***

Logistic regression was used in learning. Experimental results based on a large number of amazon.com reviews showed that type 2 and types 3 spam reviews are fairly easy to detect.

However, this cannot be said about type 1 spam, untruthful opinions or fake reviews. In fact, it is very difficult to detect such reviews because manually labeling training data is very hard, if not impossible. The problem is that identifying spam reviews by simply reading the reviews is extremely difficult because a spammer can carefully craft a spam review that is just like any innocent review.

In order to detect such spam, let us analyze fake reviews in greater detail. As indicated above, there are two main objectives for spam:

- Write undeserving positive reviews for some target objects in order to promote them. We call such spam reviews *hype spam*.
- Write unfair or malicious negative reviews for some target objects to damage their reputations. We call such spam reviews *defaming spam*.

In certain cases, the spammer may want to achieve both objectives, while in others, he/she only aims to achieve one of them because either he/she does not have an object to promote or there is no competition.

We now discuss what kinds of reviews are harmful and are likely to be spammed. Table 2 gives a simple view of type 1 spam. Spam reviews in regions 1, 3 and 5 are typically written by manufacturers of the product or persons with direct economic or other interests in the product. Their goal is to promote the product. Although opinions expressed in region 1 may be true, reviewers do not announce their conflict of interests. Note that good, bad and average products can be defined based on average review ratings given to the product. Spam reviews in regions 2, 4, and 6 are likely to be written by competitors. Although opinions in reviews of region 4 may be true, reviewers do not announce their conflict of interests and have malicious intensions. Clearly, spam reviews in region 1 and 4 are not so damaging, while spam reviews in regions 2, 3, 5 and 6 are very harmful. Thus, spam detection techniques should focus on identifying reviews in these regions. One important observation from this table is that harmful fake reviews are often outlier reviews. In other words, deviating from the norm is the necessary condition for harmful spam reviews, but not sufficient because many outlier reviews may be truthful.

Since manually labeling training data is extremely difficult, other ways have to be explored in order to find training examples for detecting possible type 1 spam. In [41], it exploits duplicate reviews. In their study of 5.8 million reviews, 2.14 million reviewers and 6.7 million products from amazon.com, they found a large number of duplicate and near-duplicate reviews, which indicates that review spam is widespread. These duplicates (which include near-duplicates) can be divided into four groups:

1. Duplicates from the same userid on the same product.
2. Duplicates from different userids on the same product.
3. Duplicates from the same userid on different products.
4. Duplicates from different userids on different products.

The first type of duplicates can be the results of reviewers mistakenly clicking the submit button multiple times (which of course can be detected based on the submission dates and times), or the same reviewers coming back to write updated reviews after using the product for some time. However, the last three kinds of duplicates are almost certainly type 1 spam reviews. Further sanity check was performed on these duplicate reviews because amazon.com cross-posts reviews to different formats of the same product, e.g., hardcover and paperback of the same book. Manually checking a large number of duplicate reviews showed that only a small percentage of them falls into this category. One reason for the low percentage could be because the reviews being studied were all from manufactured products, which perhaps have fewer formats of the same product (unlike books).

In the work reported in [41], these three types of duplicates and near duplicates are treated as type 1 spam reviews, and the rest of the reviews are treated as non-spam reviews. Logistic regression is used to build a classification model. The experiments show some tentative but interesting results.

- Negative outlier reviews (whose ratings have significant negative deviations from the average rating) tend to be heavily spammed. The reason for such spam is quite intuitive. Positive outlier reviews are not badly spammed.
- Those reviews that are the only reviews of some products are likely to be spammed. This can be explained by the tendency of promoting an unpopular product by writing a spam review.
- Top-ranked reviewers are more likely to be spammers. Amazon.com gives a rank to each member/reviewer based on the frequency that he/she gets helpful feedback on his/her reviews. Additional analysis shows that top-ranked reviewers generally write a large number of reviews. People who wrote a large number of reviews are natural suspects. Some top reviewers wrote thousands or even tens of thousands of reviews, which is unlikely for an ordinary consumer.
- Spam reviews can get good helpful feedbacks and non-spam reviews can get bad feedbacks. This is important as it shows that if usefulness or quality of a review is defined based on the helpful feedbacks that the review gets, people can be readily fooled by spam reviews. Note that the number of helpful feedbacks can be spammed too.
- Products of lower sale ranks are more likely to be spammed. This is good news because spam activities seem to be limited to low selling products, which is actually quite intuitive as it is difficult to damage the reputation of a popular product by writing a spam review.

Finally, it should be noted again that these results are only tentative because 1) it is not confirmed that the three types of duplicates are absolutely spam reviews, and 2) many spam reviews are not duplicated and they are not considered as spam in model building but are treated as non-spam due to the difficulty of manual labeling. For additional analysis and more spam detection strategies, please refer to [41]. This research is still at its infancy. Much work needs to be done. As we mentioned at the beginning of the section, with more and more people and organizations rely on opinions on the Web, devising good techniques to detect opinion spam is urgently needed. We do not want to wait until the day when the opinions on the Web are so heavily spammed that they become completely useless.

6.2 Utility of Reviews

Determining the utility of reviews is usually formulated as a regression problem. The learned model then assigns a utility value to each review, which can be used in review ranking. In this area of research, the ground truth data used for both training and testing are usually the user-helpfulness feedbacks given to each review, which as we discussed above are provided for each review at many review aggregation sites. So unlike fake review detection, the training and testing data here is not an issue.

Researchers have used many types of data features for model building [31, 49, 110]. Example features include review length, review ratings (the number of stars), counts of some specific POS tags, opinion words, tf-idf weighting scores, wh-words, product attribute mentions, product brands, comparison with product specifications, and comparison with editorial reviews, and many more. Subjectivity classification is also applied in [31]. In [57], Liu et al. formulated the problem slightly differently, as a binary classification problem. Instead of using the original helpfulness feedbacks as the classification target or dependent variable, they performed manual annotation based on whether a review comments on many product attributes/features or not.

Finally, we note again that review utility regression/classification and review spam detections are different concepts. Not-helpful or low quality reviews are not necessarily fake reviews or spam, and helpful reviews may not be non-spam. A user often determines whether a review is helpful or not based on whether the review expresses opinions on many attributes/features of the product. A spammer can satisfy this requirement by carefully crafting a review that is just like a normal helpful review. Using the number of helpful feedbacks to define review quality is also problematic because user feedbacks can be spammed too. Feedback spam is a sub-problem of click fraud in search advertising, where a person or robot clicks on some online advertisements to give the impression of real customer clicks. Here, a robot or a human spammer can also click on helpful feedback button to increase the helpfulness of a review. Another important point is that a low quality review is still a valid review and should not be discarded, but a spam review is untruthful and/or malicious and should be removed once detected.

7. Conclusions

This chapter gave an introduction to sentiment analysis and subjectivity (or opinion mining). Due to many challenging research problems and a wide variety of practical applications, it has been a very active research area in recent years. In fact, it has spread from computer science to management science [e.g., 2, 11, 17, 32, 37, 58, 74]. This chapter first presented an abstract model of sentiment analysis, which formulates the problem and provides a common framework to unify different research directions. It then discussed the most widely studied topic of sentiment and subjectivity classification, which determines whether a document or sentence is opinionated, and if so whether it carries a positive or negative opinion. We then described feature-based sentiment analysis which exploits the full power of the abstract model. After that we discussed the problem of analyzing comparative and superlative sentences. Such sentences represent a different type of evaluation from direct opinions which have been the focus of the current research. The topic of opinion search or retrieval was introduced as well, as a parallel to the general Web search. Last but not least, we discussed opinion spam, which is increasingly becoming an important issue as more and more people are relying on opinions on the Web for decision making. This gives more and more incentive for spam. There is still no effective technique to combat opinion spam.

Finally, we conclude the chapter by saying that all the sentiment analysis tasks are very challenging. Our understanding and knowledge of the problem and its solution are still limited. The main reason is that it is a natural language processing task, and natural language processing has no easy problems. Another reason may be due to our popular ways of doing research. We probably relied too much on machine learning algorithms. Some of the most effective machine learning algorithms, e.g., support vector machines and conditional random fields, produce no human understandable results such that although they may achieve improved accuracy, we know little about how and why apart from some superficial knowledge gained in the manual feature engineering process. However, that being said, we have indeed made significant progresses over the past few years. This is evident from the large number of start-up companies that offer sentiment analysis or opinion mining services. There is a real and huge need in the industry for such services because every company wants to know how consumers perceive their products and services and those of their competitors. The same can also be said about consumers because whenever one wants to buy something, one wants to know the opinions of existing users. These practical needs and the technical challenges will keep the field vibrant and lively for years to come.

Acknowledgements

I am very grateful to Theresa Wilson for her insightful and detailed comments and suggestions, which have helped me improve the chapter significantly. I thank my former and current students for working with me on this fascinating topic: Xiaowen Ding, Murthy Ganapathibhotla, Minqing Hu, Nitin Jindal, Guang Qiu (visiting student from Zhejiang University) and Lei Zhang. I would also like to express my gratitude to Birgit König (McKinsey&Company) for many valuable discussions which have helped shape my understanding of the practical side of sentiment analysis and its related issues.

References

[1]. A. Andreevskaia and S. Bergler, "Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses." Proceedings of the European Chapter of the Association for Computational Linguistics (EACL), 2006.

[2]. N. Archak, A. Ghose, and P. Ipeirotis, "Show me the money! Deriving the pricing power of product features by mining consumer reviews," Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2007.

[3]. A. Aue and M. Gamon, "Customizing sentiment classifiers to new domains: A case study," Proceedings of Recent Advances in Natural Language Processing (RANLP), 2005.

[4]. P. Beineke, T. Hastie, C. Manning, and S. Vaithyanathan. "Exploring sentiment summarization," Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text, AAAI technical report SS-04-07, 2004.

[5]. S. Bethard, H. Yu, A. Thornton, V. Hatzivassiloglou, and D. Jurafsky, "Automatic extraction of opinion propositions and their holders," Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text, 2004.

[6]. J. Blitzer, M. Dredze, and F. Pereira, "Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," Proceedings of the Association for Computational Linguistics (ACL), 2007.

[7]. J. Bos, and M. Nissim, "An Empirical Approach to the Interpretation of Superlatives." Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2006.

[8]. E. Breck, Y. Choi, and C. Cardie, "Identifying expressions of opinion in context," Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2007.

[9]. G. Carenini, R. Ng, and A. Pauls, "Multi-document summarization of evaluative text," Proceedings of the European Chapter of the Association for Computational Linguistics (EACL), pp. 305–312, 2006.

[10]. G. Carenini, R. T. Ng, and E. Zwart, "Extracting knowledge from evaluative text," Proceedings of International Conference on Knowledge Capture (K-CAP), pp. 11-18, 2005.

[11]. Y. Chen and J. Xie, "Online consumer review: Word-of-mouth as a new element of marketing communication mix," Management Science, vol. 54, pp. 477–491, 2008.

[12]. P. Chesley, B. Vincent, L. Xu, and R. Srihari, "Using verbs and adjectives to automatically classify blog sentiment," in AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW), pp. 27–29, 2006.

[13]. Y. Choi, E. Breck, and C. Cardie, "Joint extraction of entities and relations for opinion recognition," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2006.

[14]. Y. Choi, C. Cardie, E. Riloff, and S. Patwardhan, "Identifying sources of opinions with conditional random fields and extraction patterns," Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), 2005.

[15]. S. R. Das and M. Y. Chen, "Yahoo! for Amazon: Sentiment extraction from small talk on the Web," Management Science, vol. 53, pp. 1375–1388, 2007.

[16]. K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews," Proceedings of WWW, pp. 519–528, 2003.

[17]. C. Dellarocas, X. Zhang, and N. F. Awad, "Exploring the value of online product ratings in revenue forecasting: The case of motion pictures," Journal of Interactive Marketing, vol. 21, pp. 23–45, 2007.

[18]. A. Devitt and K. Ahmad, "Sentiment analysis in financial news: A cohesion based approach," Proceedings of the Association for Computational Linguistics (ACL), pp. 984–991, 2007.

[19]. X. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," Proceedings of the Conference on Web Search and Web Data Mining (WSDM), 2008.

[20]. K. Eguchi and V. Lavrenko, “Sentiment retrieval using generative models,” Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 345–354, 2006.
[21]. A. Esuli and F. Sebastiani, “Determining the semantic orientation of terms through gloss analysis,” Proceedings of the ACM Conference on Information and Knowledge Management (CIKM), 2005.
[22]. A. Esuli and F. Sebastiani, “Determining term subjectivity and term orientation for opinion mining,” Proceedings of the European Chapter of the Association for Computational Linguistics (EACL), 2006.
[23]. A. Esuli and F. Sebastiani, “SentiWordNet: A publicly available lexical resource for opinion mining,” Proceedings of Language Resources and Evaluation (LREC), 2006.
[24]. A. Esuli and F. Sebastiani, “PageRanking WordNet synsets: An application to opinion mining,” Proceedings of the Association for Computational Linguistics (ACL), 2007.
[25]. C. Fellbaum, ed., Wordnet: An Electronic Lexical Database. MIT Press, 1998.
[26]. M. Fiszman, D. Demner-Fushman, F. Lang, P. Goetz and T. Rindflesch, “Interpreting comparative constructions in biomedical text,” BioNLP, 2007.
[27]. M. Gamon, “Sentiment classification on customer feedback data: Noisy data, large feature vectors, and the role of linguistic analysis,” Proceedings of the International Conference on Computational Linguistics (COLING), 2004.
[28]. M. Gamon, A. Aue, S. Corston-Oliver, and E. Ringger, “Pulse: Mining customer opinions from free text,” Proceedings of the International Symposium on Intelligent Data Analysis (IDA), pp. 121–132, 2005.
[29]. G. Ganapathibhotla and B. Liu. “Identifying Preferred Entities in Comparative Sentences,” Proceedings of the International Conference on Computational Linguistics, COLING, 2008.
[30]. R. Ghani, K. Probst, Y. Liu, M. Krema, and A. Fano, “Text mining for product attribute extraction,” SIGKDD Explorations Newsletter, vol. 8, pp. 41–48, 2006.
[31]. A. Ghose and P. G. Ipeirotis, “Designing novel review ranking systems: Predicting usefulness and impact of reviews,” Proceedings of the International Conference on Electronic Commerce (ICEC), 2007.
[32]. A. Ghose, P. G. Ipeirotis, and A. Sundararajan, “Opinion mining using econometrics: A case study on reputation systems,” Proceedings of the Association for Computational Linguistics (ACL), 2007.
[33]. N. Godbole, M. Srinivasaiah, and S. Skiena, “Large-scale sentiment analysis for news and blogs,” Proceedings of the International Conference on Weblogs and Social Media (ICWSM), 2007.
[34]. V. Hatzivassiloglou and K. McKeown, “Predicting the semantic orientation of adjectives,” Proceedings of the Joint ACL/EACL Conference, pp. 174–181, 1997.
[35]. V. Hatzivassiloglou and J. Wiebe, “Effects of adjective orientation and gradability on sentence subjectivity,” Proceedings of the International Conference on Computational Linguistics (COLING), 2000.
[36]. M. Hu and B. Liu, “Mining and summarizing customer reviews,” Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), pp. 168–177, 2004.
[37]. N. Hu, P. A. Pavlou, and J. Zhang, “Can online reviews reveal a product’s true quality?: Empirical findings and analytical modeling of online word-of-mouth communication,” Proceedings of Electronic Commerce (EC), pp. 324–330, USA, New York, NY: ACM, 2006.
[38]. N. Jindal and B. Liu, “Identifying comparative sentences in text documents,” Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR), 2006.
[39]. N. Jindal and B. Liu, “Mining comparative sentences and relations,” Proceedings of AAAI, 2006.
[40]. N. Jindal and B. Liu, “Review spam detection,” Proceedings of WWW, 2007. (Poster paper).
[41]. N. Jindal and B. Liu, “Opinion spam and analysis,” Proceedings of the Conference on Web Search and Web Data Mining (WSDM), pp. 219–230, 2008.
[42]. N. Kaji and M. Kitsuregawa, “Building lexicon for sentiment analysis from massive collection of HTML documents,” Proceedings of the Joint Conference on Empirical Methods in Natural

Language Processing and ComputationalNatural Language Learning (EMNLP-CoNLL), pp. 1075–1083, 2007.
[43]. J. Kamps, M. Marx, R. J. Mokken and M. de Rijke. *Using WordNet to measure semantic orientation of adjectives.* In Proc. of LREC'04, pp. 1115-1118, 2004.
[44]. H. Kanayama and T. Nasukawa, "Fully automatic lexicon expansion for domain-oriented sentiment analysis," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 355–363, July 2006.
[45]. A. Kennedy and D. Inkpen, "Sentiment classification of movie reviews using contextual valence shifters," Computational Intelligence, vol. 22, pp. 110–125, 2006.
[46]. S.-M. Kim and E. Hovy, "Determining the sentiment of opinions," Proceedings of the International Conference on Computational Linguistics (COLING), 2004.
[47]. S.-M. Kim and E. Hovy, "Automatic identification of pro and con reasons in online reviews," Proceedings of the COLING/ACL Main Conference Poster Sessions, pp. 483–490, 2006.
[48]. S.-M. Kim and E. Hovy, "Crystal: Analyzing predictive opinions on the web," Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP/CoNLL), 2007.
[49]. S.-M. Kim, P. Pantel, T. Chklovski, and M. Pennacchiotti, "Automatically assessing review helpfulness," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 423–430, Sydney, Australia, July 2006.
[50]. N. Kobayashi, K. Inui and Y. Matsumoto. Extracting Aspect-Evaluation and Aspect-of Relations in Opinion Mining**.** Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1065–1074, 2007.
[51]. N. Kobayashi, K. Inui, Y. Matsumoto, K. Tateishi, and T. Fukushima. Collecting evaluative expressions for opinion extraction. Proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP), pages 584–589, 2004.
[52]. L.-W. Ku, Y.-T. Liang, and H.-H. Chen, "Opinion extraction, summarization and tracking in news and blog corpora," in AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW), pp. 100–107, 2006.
[53]. L.-W. Ku, Y.-T. Liang, and H.-H. Chen, Novel Relationship Discovery Using Opinions Mined from the Web. AAAI 2006.
[54]. J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," Proceedings of ICML, pp. 282–289, 2001.
[55]. B. Liu, Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data. Springer, 2006.
[56]. B. Liu, M. Hu, and J. Cheng, "Opinion observer: Analyzing and comparing opinions on the web," Proceedings of WWW, 2005.
[57]. J. Liu, Y. Cao, C.-Y. Lin, Y. Huang, and M. Zhou. "Low-quality product review detection in opinion summarization," Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 334–342, 2007. (Poster paper).
[58]. Y. Liu, J. Huang, A. An, and X. Yu. "ARSA: A sentiment-aware model for predicting sales performance using blogs," Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR), 2007.
[59]. Y. Lu and C. X. Zhai. Opinion integration through semi-supervised topic modeling. Proceedings of 2008 International WWW Conference (WWW'08), pp. 121–130, 2008.
[60]. C. MacDonald, I. Ounis, and I. Soboroff. Overview of the TREC2007 Blog Track. 2007. http://trec.nist.gov/pubs/trec16/papers/BLOG.OVERVIEW16.pdf.
[61]. J. Martin and P. White. *The Language of Evaluation, Appraisal in English*, Palgrave Macmillan, London, New York, 2005.
[62]. R. McDonald, K. Hannan, T. Neylon, M. Wells, and J. Reynar, "Structured models for fine-to-coarse sentiment analysis," Proceedings of the Association for Computational Linguistics (ACL), pp. 432–439, Prague, Czech Republic: June 2007.

[63]. Q. Mei, X. Ling, M. Wondra, H. Su, and C. X. Zhai, “Topic sentiment mixture: Modeling facets and opinions in weblogs,” Proceedings of WWW, pp. 171–180, 2007.
[64]. R. Mihalcea, C. Banea, and J. Wiebe, “Learning multilingual subjective language via cross-lingual projections,” Proceedings of the Association for Computational Linguistics (ACL), pp. 976–983, 2007.
[65]. S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima, “Mining product reputations on the Web,” Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), pp. 341–349, 2002. (Industry track).
[66]. T. Nasukawa and J. Yi, “Sentiment analysis: Capturing favorability using natural language processing,” Proceedings of the Conference on Knowledge Capture (K-CAP), 2003.
[67]. V. Ng, S. Dasgupta, and S. M. N. Arifin, “Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews,” Proceedings of the COLING/ACL Main Conference Poster Sessions, pp. 611–618, Sydney, Australia: July 2006.
[68]. X. Ni, G.-R. Xue, X. Ling, Y. Yu, and Q. Yang, “Exploring in the weblog space by detecting informative and affective articles,” Proceedings of WWW, 2007.
[69]. K. Nigam and M. Hurst, “Towards a robust metric of polarity,” in Computing Attitude and Affect in Text: Theories and Applications, number 20 in The Information Retrieval Series, (J. G. Shanahan, Y. Qu, and J. Wiebe, eds.), 2006.
[70]. B. Pang and L. Lee, “A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts,” Proceedings of the Association for Computational Linguistics (ACL), pp. 271–278, 2004.
[71]. B. Pang and L. Lee, “Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales,” Proceedings of the Association for Computational Linguistics (ACL), pp. 115–124, 2005.
[72]. B. Pang and L. Lee, “Opinion mining and sentiment analysis.” Foundations and Trends in Information Retrieval 2(1-2), pp. 1–135, 2008.
[73]. B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? Sentiment classification using machine learning techniques,” Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 79–86, 2002.
[74]. D.-H. Park, J. Lee, and I. Han, “The effect of on-line consumer reviews on consumer purchasing intention: The moderating role of involvement,” International Journal of Electronic Commerce, vol. 11, pp. 125–148, 2007.
[75]. W. Parrott. Emotions in Social Psychology, Psychology Press, Philadelphia, 2001.
[76]. L. Polanyi and A. Zaenen, “Contextual lexical valence shifters,” Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text, 2004.
[77]. A.-M. Popescu and O. Etzioni, “Extracting product features and opinions from reviews,” Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), 2005.
[78]. G. Qiu, B. Liu, J. Bu and C. Chen. Expanding Domain Sentiment Lexicon through Double Propagation, International Joint Conference on Artificial Intelligence (IJCAI-09), 2009.
[79]. E. Riloff, S. Patwardhan, and J. Wiebe, “Feature subsumption for opinion analysis,” Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2006.
[80]. E. Riloff and J. Wiebe, “Learning extraction patterns for subjective expressions,” Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2003.
[81]. E. Riloff, J. Wiebe, and T. Wilson, “Learning subjective nouns using extraction pattern bootstrapping,” Proceedings of the Conference on Natural Language Learning (CoNLL), pp. 25–32, 2003.
[82]. S. Sarawagi, “Information extraction,” to appear in Foundations and Trends in Information Retrieval, 2009.
[83]. Y. Seki, K. Eguchi, N. Kando, and M. Aono, “Opinion-focused summarization and its analysis at DUC 2006,” Proceedings of the Document Understanding Conference (DUC), pp. 122–130, 2006.

[84]. B. Snyder and R. Barzilay. Multiple Aspect Ranking Using the Good Grief Algorithm. Proceedings of Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, 2007 HLT-NAACL 2007: 300-307.
[85]. X. Song, Y. Chi, K. Hino, and B. Tseng, "Identifying opinion leaders in the blogosphere," Proceedings of the Conference on Information and Knowledge Management (CIKM), pp. 971–974, 2007.
[86]. A. Stepinski and V. Mittal, "A fact/opinion classifier for news articles," Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR), pp. 807–808, 2007.
[87]. P. J. Stone. The General Inquirer: A Computer Approach to Content Analysis. The MIT Press, 1966.
[88]. V. Stoyanov and C. Cardie, "Partially supervised coreference resolution for opinion summarization through structured rule learning," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 336–344, Sydney, Australia: July 2006.
[89]. Q. Su, X. Xu, H. Guo, X. Wu, X. Zhang, B. Swen, and Z. Su. Hidden Sentiment Association in Chinese Web Opinion Mining. *Proceedings of WWW'08*, pp. 959-968, 2008.
[90]. H. Takamura, T. Inui, and M. Okumura, "Extracting semantic orientations of phrases from dictionary," Proceedings of the Joint Human Language Technology/North American Chapter of the ACL Conference (HLT-NAACL), 2007.
[91]. L. Tesnière, éléments de syntaxe structurale, Klincksieck, Paris 1959.
[92]. M. Thomas, B. Pang, and L. Lee, "Get out the vote: Determining support or opposition from congressional floor-debate transcripts," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 327–335, 2006.
[93]. I. Titov and R. McDonald. A Joint Model of Text and Aspect Ratings for Sentiment Summarization. Proceedings of 46th Annual Meeting of the Association for Computational Linguistics (ACL'08), 2008.
[94]. R. M. Tong, "An operational system for detecting and tracking opinions in on-line discussion." Proceedings of the Workshop on Operational Text Classification (OTC), 2001.
[95]. P. Turney, "Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews," Proceedings of the Association for Computational Linguistics (ACL), pp. 417–424, 2002.
[96]. X. Wan. "Using bilingual knowledge and ensemble techniques for unsupervised Chinese sentiment analysis." In Proceedings of EMNLP08, pp. 553-561, 2008.
[97]. J. Wiebe, "Learning subjective adjectives from corpora," Proceedings of AAAI, 2000.
[98]. J. Wiebe, R. F. Bruce, and T. P. O'Hara. "Development and use of a gold standard data set for subjectivity classifications." Proceedings of the Association for Computational Linguistics (ACL), pp. 246–253, 1999.
[99]. J. Wiebe and R. Mihalcea. "Word sense and subjectivity." Proceedings of the Conference on Computational Linguistics / Association for Computational Linguistics (COLING/ACL), 2006.
[100]. J. Wiebe and T. Wilson, "Learning to disambiguate potentially subjective expressions," Proceedings of the Conference on Natural Language Learning (CoNLL), pp. 112–118, 2002.
[101]. J. Wiebe, T. Wilson and C. Cardie. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 1(2), 2005.
[102]. J. Wiebe, T. Wilson, R. Bruce, M. Bell, and M. Martin, "Learning subjective language," Computational Linguistics, vol. 30, pp. 277–308, September 2004.
[103]. T. Wilson, J. Wiebe, and R. Hwa, "Just how mad are you? Finding strong and weak opinion clauses," Proceedings of AAAI, pp. 761–769, 2004.
[104]. T. Wilson, J. Wiebe, and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis." Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pp. 347–354, 2005.
[105]. H. Yang, L. Si, and J. Callan, "Knowledge transfer and opinion detection in the TREC2006 blog track," Proceedings of TREC, 2006.

[106]. J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack, “Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques,” Proceedings of the IEEE International Conference on Data Mining (ICDM), 2003.
[107]. H. Yu and V. Hatzivassiloglou, “Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences,” Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2003.
[108]. W. Zhang, L. Jia, C. Yu, W. Meng. Improve the Effectiveness of the Opinion Retrieval and Opinion Polarity Classification. ACM 17th Conference on Information and Knowledge Management (CIKM 2008), 2008.
[109]. W. Zhang, and C. Yu. UIC at TREC 2007 Blog Report., 2007. http://trec.nist.gov/pubs/trec16/papers/uic-zhang.blog.final.pdf
[110]. Z. Zhang and B. Varadarajan, “Utility scoring of product reviews,” Proceedings of the ACM Conference on Information and Knowledge Management (CIKM), pp. 51–57, 2006.
[111]. L. Zhuang, F. Jing, X.-Y. Zhu, and L. Zhang, “Movie review mining and summarization,” Proceedings of the ACM Conference on Information and Knowledge Management (CIKM), 2006.