

# Automatic Construction of Polarity-tagged Corpus from HTML Documents

Nobuhiro Kaji and Masaru Kitsuregawa

Institute of Industrial Science

the University of Tokyo

4-6-1 Komaba, Meguro-ku, Tokyo 153-8505 Japan

{kaji,kitsure}@tkl.iis.u-tokyo.ac.jp

## Abstract

This paper proposes a novel method of building polarity-tagged corpus from HTML documents. The characteristics of this method is that it is fully automatic and can be applied to arbitrary HTML documents. The idea behind our method is to utilize certain layout structures and linguistic pattern. By using them, we can automatically extract such sentences that express opinion. In our experiment, the method could construct a corpus consisting of 126,610 sentences.

## 1 Introduction

Recently, there has been an increasing interest in such applications that deal with opinions (a.k.a. sentiment, reputation etc.). For instance, Morinaga et al. developed a system that extracts and analyzes reputations on the Internet (Morinaga et al., 2002). Pang et al. proposed a method of classifying movie reviews into positive and negative ones (Pang et al., 2002).

In these applications, one of the most important issue is how to determine the *polarity* (or *semantic orientation*) of a given text. In other words, it is necessary to decide whether a given text conveys positive or negative content.

In order to solve this problem, we intend to take statistical approach. More specifically, we plan to learn the polarity of texts from a corpus in which phrases, sentences or documents are tagged with labels expressing the polarity (*polarity-tagged corpus*).

So far, this approach has been taken by a lot of researchers (Pang et al., 2002; Dave et al., 2003; Wilson et al., 2005). In these previous works,

polarity-tagged corpus was built in either of the following two ways. It is built manually, or created from review sites such as AMAZON.COM. In some review sites, the review is associated with metadata indicating its polarity. Those reviews can be used as polarity-tagged corpus. In case of AMAZON.COM, the review's polarity is represented by using 5-star scale.

However, both of the two approaches are not appropriate for building large polarity-tagged corpus. Since manual construction of tagged corpus is time-consuming and expensive, it is difficult to build large polarity-tagged corpus. The method that relies on review sites can not be applied to domains in which large amount of reviews are not available. In addition, the corpus created from reviews is often noisy as we discuss in Section 2.

This paper proposes a novel method of building polarity-tagged corpus from HTML documents. The idea behind our method is to utilize certain layout structures and linguistic pattern. By using them, we can automatically extract sentences that express opinion (opinion sentences) from HTML documents. Because this method is fully automatic and can be applied to arbitrary HTML documents, it does not suffer from the same problems as the previous methods.

In the experiment, we could construct a corpus consisting of 126,610 sentences. To validate the quality of the corpus, two human judges assessed a part of the corpus and found that 92% opinion sentences are appropriate ones. Furthermore, we applied our corpus to opinion sentence classification task. Naive Bayes classifier was trained on our corpus and tested on three data sets. The result demonstrated that the classifier achieved more than 80% accuracy in each data set.

The following of this paper is organized as fol-

lows. Section 2 shows the design of the corpus constructed by our method. Section 3 gives an overview of our method, and the detail follows in Section 4. In Section 5, we discuss experimental results, and in Section 6 we examine related works. Finally we conclude in Section 7.

## 2 Corpus Design

This Section explains the design of our corpus that is built automatically. Table 1 represents a part of our corpus that was actually constructed in the experiment. Note that this paper treats Japanese. The sentences in the Table are translations, and the original sentences are in Japanese.

The followings are characteristics of our corpus:

- Our corpus uses two labels, + and -. They denote positive and negative sentences respectively. Other labels such as 'neutral' are not used.
- Since we do not use 'neutral' label, such sentence that does not convey opinion is not stored in our corpus.
- The label is assigned to not multiple sentences (or document) but single sentence. Namely, our corpus is tagged at sentence level rather than document level.

It is important to discuss the reason that we intend to build a corpus tagged at sentence level rather than document level. The reason is that one document often includes both positive and negative sentences, and hence it is difficult to learn the polarity from the corpus tagged at document level. Consider the following example (Pang et al., 2002):

This film should be brilliant. It sounds like a great plot, the actors are first grade, and the supporting cast is good as well, and Stallone is attempting to deliver a good performance. However, it can't hold up.

This document as a whole expresses negative opinion, and should be labeled 'negative' if it is tagged at document level. However, it includes several sentences that represent positive attitude.

We would like to point out that polarity-tagged corpus created from reviews prone to be tagged at document-level. This is because meta-data (e.g. stars in AMAZON.COM) is usually associated with

one review rather than individual sentences in a review. This is one serious problem in previous works.

Table 1: A part of automatically constructed polarity-tagged corpus.

label	opinion sentence
+	It has high adaptability.
-	The cost is expensive.
-	The engine is powerless and noisy.
+	The usage is easy to understand.
+	Above all, the price is reasonable.

## 3 The Idea

This Section briefly explains our basic idea, and the detail of our corpus construction method is represented in the next Section.

Our idea is to use certain layout structures and linguistic pattern in order to extract opinion sentences from HTML documents. More specifically, we used two kinds of layout structures: the itemization and the table. In what follows, we explain examples where opinion sentences can be extracted by using the itemization, table and linguistic pattern.

### 3.1 Itemization

The first idea is to extract opinion sentences from the itemization (Figure 1). In this Figure, opinions about a music player are itemized and these itemizations have headers such as 'pros' and 'cons'. By using the headers, we can recognize that opinion sentences are described in these itemizations.

<p><b>Pros:</b></p> <ul style="list-style-type: none"> <li>• The sound is natural.</li> <li>• Music is easy to find.</li> <li>• Can enjoy creating my favorite play-lists.</li> </ul> <p><b>Cons:</b></p> <ul style="list-style-type: none"> <li>• The remote controller does not have an LCD display.</li> <li>• The body gets scratched and fingerprinted easily.</li> <li>• The battery drains quickly when using the back-light.</li> </ul>
---

Figure 1: Opinion sentences in itemization.

Hereafter, such phrases that indicate the pres-

ence of opinion sentences are called *indicators*. Indicators for positive sentences are called *positive indicators*. 'Pros' is an example of positive indicator. Similarly, indicators for negative sentences are called *negative indicators*.

### 3.2 Table

The second idea is to use the table structure (Figure 2). In this Figure, a car review is summarized in the table.

Mileage(urban)	7.0km/litter
Mileage(highway)	9.0km/litter
Plus	This is a four door car, but it's so cool.
Minus	The seat is ragged and the light is dark.

Figure 2: Opinion sentences in table.

We can predict that there are opinion sentences in this table, because the left column acts as a header and there are indicators (plus and minus) in that column.

### 3.3 Linguistic pattern

The third idea is based on linguistic pattern. Because we treat Japanese, the pattern that is discussed in this paper depends on Japanese grammar although we think there are similar patterns in other languages including English.

Consider the Japanese sentences attached with English translations (Figure 3). Japanese sentences are written in italics and '-' denotes that the word is followed by postpositional particles. For example, '*software-no*' means that '*software*' is followed by postpositional particle '*no*'. Translations of each word and the entire sentence are represented below the original Japanese sentence. '-POST' means postpositional particle.

In the examples, we focused on the singly underlined phrases. Roughly speaking, they correspond to 'the advantage/weakness is to' in English. In these phrases, indicators ('*riten* (advantage)' and '*ketten* (weakness)') are followed by postpositional particle '*-ha*', which is topic marker. And hence, we can recognize that something good (or bad) is the topic of the sentence.

Based on this observation, we crafted a linguistic pattern that can detect the singly underlined phrases. And then, we extracted doubly underlined phrases as opinions. They correspond to 'run quickly' and 'take too much time'. The detail of this process is discussed in the next Section.

## 4 Automatic Corpus Construction

This Section represents the detail of the corpus construction procedure.

As shown in the previous Section, our idea utilizes the indicator, and it is important to recognize indicators in HTML documents. To do this, we manually crafted lexicon, in which positive and negative indicators are listed. This lexicon consists of 303 positive and 433 negative indicators.

Using this lexicon, the polarity-tagged corpus is constructed from HTML documents. The method consists of the following three steps:

### 1. Preprocessing

Before extracting opinion sentences, HTML documents are preprocessed. This process involves separating texts from HTML tags, recognizing sentence boundary, and complementing omitted HTML tags etc.

### 2. Opinion sentence extraction

Opinion sentences are extracted from HTML documents by using the itemization, table and linguistic pattern.

### 3. Filtering

Since HTML documents are noisy, some of the extracted opinion sentences are not appropriate. They are removed in this step.

For the preprocessing, we implemented simple rule-based system. We cannot explain its detail for lack of space. In the remainder of this Section, we describe three extraction methods respectively, and then examine filtering technique.

### 4.1 Extraction based on itemization

The first method utilizes the itemization. In order to extract opinion sentences, first of all, we have to find such itemization as illustrated in Figure 1. They are detected by using indicator lexicon and HTML tags such as <h1> and <ul> etc.

After finding the itemizations, the sentences in the items are extracted as opinion sentences. Their polarity labels are assigned according to whether the header is positive or negative indicator. From the itemization in Figure 1, three positive sentences and three negative ones are extracted.

The problem here is how to treat such item that has more than one sentences (Figure 4). In this itemization, there are two sentences in each of the

- (1) *kono software-no riten-ha hayaku ugoku koto*  
 this software-POST advantage-POST quickly run to  
The advantage of this software is to run quickly.
- (2) *ketten-ha jikan-ga kakarisugiru koto-desu*  
 weakness-POST time-POST take too much to-POST  
The weakness is to take too much time.

Figure 3: Instances of the linguistic pattern.

third and fourth item. It is hard to precisely predict the polarity of each sentence in such items, because such item sometimes includes both positive and negative sentences. For example, in the third item of the Figure, there are two sentences. One ('Has high pixel...') is positive and the other ('I was not satisfied...') is negative.

To get around this problem, we did not use such items. From the itemization in Figure 4, only two positive sentences are extracted ('the color is really good' and 'this camera makes me happy while taking pictures').

<b>Pros:</b>
<ul style="list-style-type: none"> <li>• The color is really good.</li> <li>• This camera makes me happy while taking pictures.</li> <li>• Has high pixel resolution with 4 million pixels. I was not satisfied with 2 million.</li> <li>• EVF is easy to see. But, compared with SLR, it's hard to see.</li> </ul>

Figure 4: Itemization where more than one sentences are written in one item.

## 4.2 Extraction based on table

The second method extracts opinion sentences from the table. Since the combination of <table> and other tags can represent various kinds of tables, it is difficult to craft precise rules that can deal with any table.

Therefore, we consider only two types of tables in which opinion sentences are described (Figure 5). Type A is a table in which the leftmost column acts as a header, and there are indicators in that column. Similarly, type B is a table in which the first row acts as a header. The table illustrated in Figure 2 is categorized into type A.

The type of the table is decided as follows. The table is categorized into type A if there are both

type A			
$I_+$	+	+	+
$I_-$	-	-	-

type B			
	$I_+$	$I_-$	
	+	-	
	+	-	
	+	-	

$I_+$ :positive indicator    +:positive sentence  
 $I_-$ :negative indicator    -:negative sentence

Figure 5: Two types of tables.

positive and negative indicators in the leftmost column. The table is categorized into type B if it is not type A and there are both positive and negative indicators in the first row.

After the type of the table is decided, we can extract opinion sentences from the cells that correspond to + and - in the Figure 5. It is obvious which label (positive or negative) should be assigned to the extracted sentence.

We did not use such cell that contains more than one sentences, because it is difficult to reliably predict the polarity of each sentence. This is similar to the extraction from the itemization.

## 4.3 Extraction based on linguistic pattern

The third method uses linguistic pattern. The characteristic of this pattern is that it takes dependency structure into consideration.

First of all, we explain Japanese dependency structure. Figure 6 depicts the dependency representations of the sentences in the Figure 3. Japanese sentence is represented by a set of dependencies between phrasal units called *bunsetsu*-phrases. Broadly speaking, *bunsetsu*-phrase is an unit similar to baseNP in English. In the Figure, square brackets enclose *bunsetsu*-phrase and arrows show modifier  $\rightarrow$  head dependencies between *bunsetsu*-phrases.

In order to extract opinion sentences from these dependency representations, we crafted the following dependency pattern.

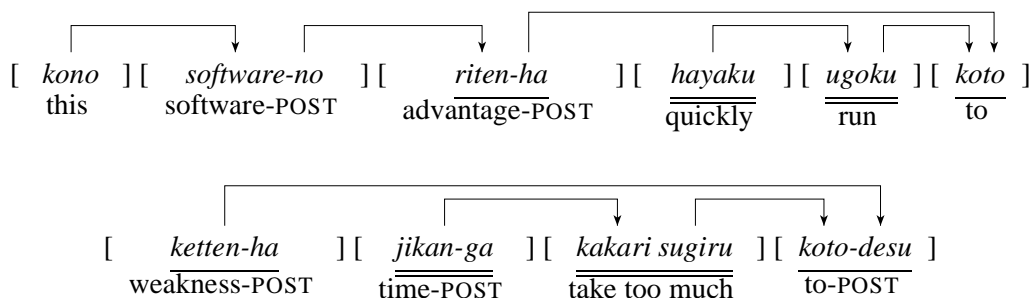


Figure 6: Dependency representations.

[ INDICATOR-*ha* ] [ *koto*-POST\* ]

This pattern matches the singly underlined *bunsetsu*-phrases in the Figure 6. In the modifier part of this pattern, the indicator is followed by postpositional particle 'ha', which is topic marker<sup>1</sup>. In the head part, 'koto (to)' is followed by arbitrary numbers of postpositional particles.

If we find the dependency that matches this pattern, a phrase between the two *bunsetsu*-phrases is extracted as opinion sentence. In the Figure 6, the doubly underlined phrases are extracted. This heuristics is based on Japanese word order constraint.

#### 4.4 Filtering

Sentences extracted by the above methods sometimes include noise text. Such texts have to be filtered out. There are two cases that need filtering process.

First, some of the extracted sentences do not express opinions. Instead, they represent objects to which the writer's opinion is directed (Table 7). From this table, 'the overall shape' and 'the shape of the taillight' are wrongly extracted as opinion sentences. Since most of the objects are noun phrases, we removed such sentences that have the noun as the head.

Mileage(urban)	10.0km/litter
Mileage(highway)	12.0km/litter
Plus	The overall shape.
Minus	The shape of the taillight.

Figure 7: A table describing only objects to which the opinion is directed.

Secondly, we have to treat duplicate opinion sentences because there are mirror sites in the

<sup>1</sup>To be exact, some of the indicators such as 'strong point' consists of more than one *bunsetsu*-phrase, and the modifier part sometimes consists of more than one *bunsetsu*-phrase.

HTML documents. When there are more than one sentences that are exactly the same, one of them is held and the others are removed.

## 5 Experimental Results and Discussion

This Section examines the results of corpus construction experiment. To analyze Japanese sentence we used Juman and KNP<sup>2</sup>.

### 5.1 Corpus Construction

About 120 millions HTML documents were processed, and 126,610 opinion sentences were extracted. Before the filtering, there were 224,002 sentences in our corpus. Table 2 shows the statistics of our corpus. The first column represents the three extraction methods. The second and third column shows the number of positive and negative sentences by extracted each method. Some examples are illustrated in Table 3.

Table 2: # of sentences in the corpus.

	Positive	Negative	Total
Itemization	18,575	15,327	33,902
Table	12,103	11,016	23,119
Linguistic Pattern	34,282	35,307	69,589
Total	64,960	61,650	126,610

The result revealed that more than half of the sentences are extracted by linguistic pattern (see the fourth row). Our method turned out to be effective even in the case where only plain texts are available.

### 5.2 Quality assessment

In order to check the quality of our corpus, 500 sentences were randomly picked up and two judges manually assessed whether appropriate labels are assigned to the sentences.

The evaluation procedure is the followings.

<sup>2</sup><http://www.kc.t.u-tokyo.ac.jp/nl-resource/top.html>

Table 3: Examples of opinion sentences.

label	opinion sentence			
+	<i>cost</i>	<i>keisan-ga</i>	<i>yoininaru</i>	
	cost	computation-POST	become easy	
	It becomes easy to compute cost.			
+	<i>kantan-de</i>	<i>jikan-ga</i>	<i>setsuyakudekiru</i>	
	easy-POST	time-POST	can save	
	It's easy and can save time.			
+	<i>soup-ha</i>	<i>koku-ga ari</i>	<i>oishii</i>	
	soup-POST	rich	flavorful	
	The soup is rich and flavorful.			
-	<i>HTML</i>	<i>keishiki-no</i>	<i>mail-ni</i>	<i>taioshitena</i>
	HTML	format-POST	mail-POST	cannot use
	Cannot use mails in HTML format.			
-	<i>jugyo-ga</i>	<i>hijoni</i>	<i>tsumaranai</i>	
	lecture-POST	really	boring	
	The lecture is really boring.			
-	<i>kokoro-ni nokoru</i>	<i>ongaku-ga</i>	<i>nai</i>	
	impressive	music-POST	there is no	
	There is no impressive music.			

- Each of the 500 sentences are shown to the two judges. Throughout this evaluation, We did not present the label automatically tagged by our method. Similarly, we did not show HTML documents from which the opinion sentences are extracted.
- The two judges individually categorized each sentence into three groups: positive, negative and neutral/ambiguous. The sentence is classified into the third group, if it does not express opinion (neutral) or if its polarity depends on the context (ambiguous). Thus, two goldstandard sets were created.
- The precision is estimated using the goldstandard. In this evaluation, the precision refers to the ratio of sentences where correct labels are assigned by our method. Since we have two goldstandard sets, we can report two different precision values. A sentence that is categorized into neutral/ambiguous by the judge is interpreted as being assigned incorrect label by our method, since our corpus does not have a label that corresponds to neutral/ambiguous.

We investigated the two goldstandard sets, and found that the judges agree with each other in 467 out of 500 sentences (93.4%). The Kappa value was 0.901. From this result, we can say that the goldstandard was reliably created by the judges.

Then, we estimated the precision. The precision was 459/500 (91.5%) when one goldstandard was used, and 460/500 (92%) when the other was used.

Since these values are nearly equal to the agreement between humans (467/500), we can conclude that our method successfully constructed polarity-tagged corpus.

After the evaluation, we analyzed errors and found that most of them were caused by the lack of context. The following is a typical example.

You see, there is much information.

In our corpus this sentence is categorized into positive one. The below is a part of the original document from which this sentence was extracted.

I recommend this guide book. The Pros. of this book is that, *you see, there is much information.*

On the other hand, both of the two judges categorized the above sentence into neutral/ambiguous, probably because they can easily assume context where much information is not desirable.

*You see, there is much information.* But, it is not at all arranged, and makes me confused.

In order to precisely treat this kind of sentences, we think discourse analysis is inevitable.

### 5.3 Application to opinion classification

Next, we applied our corpus to opinion sentence classification. This is a task of classifying sentences into positive and negative. We trained a classifier on our corpus and investigated the result.

**Classifier and data sets** As a classifier, we chose Naive Bayes with bag-of-words features, because it is one of the most popular one in this task. Negation was processed in a similar way as previous works (Pang et al., 2002).

To validate the accuracy of the classifier, three data sets were created from review pages in which the review is associated with meta-data. To build data sets tagged at sentence level, we used such reviews that contain only one sentence. Table 4 represents the domains and the number of sentences in each data set. Note that we confirmed there is no duplicate between our corpus and the these data sets.

**The result and discussion** Naive Bayes classifier was trained on our corpus and tested on the three data sets (Table 5). In the Table, the second column represents the accuracy of the classification in each data set. The third and fourth

Table 5: Classification result.

	Accuracy	Positive		Negative	
		Precision	Recall	Precision	Recall
Computer	0.831	0.856	0.804	0.804	0.859
Restaurant	0.849	0.905	0.859	0.759	0.832
Car	0.833	0.860	0.844	0.799	0.819

Table 4: The data sets.

Domain	# of sentences	
	Positive	Negative
Computer	933	910
Restaurant	753	409
Car	1,056	800

columns represent precision and recall of positive sentences. The remaining two columns show those of negative sentences. Naive Bayes achieved over 80% accuracy in all the three domains.

In order to compare our corpus with a small domain specific corpus, we estimated accuracy in each data set using 10 fold crossvalidation (Table 6). In two domains, the result of our corpus outperformed that of the crossvalidation. In the other domain, our corpus is slightly better than the crossvalidation.

Table 6: Accuracy comparison.

	Our corpus	Crossvalidation
	Computer	0.831
Restaurant	0.849	0.848
Car	0.833	0.808

One finding is that our corpus achieved good accuracy, although it includes various domains and is not accustomed to the target domain. Turney also reported good result without domain customization (Turney, 2002). We think these results can be further improved by domain adaptation technique, and it is one future work.

Furthermore, we examined the variance of the accuracy between different domains. We trained Naive Bayes on each data set and investigate the accuracy in the other data sets (Table 7). For example, when the classifier is trained on Computer and tested on Restaurant, the accuracy was 0.757. This result revealed that the accuracy is quite poor when the training and test sets are in different domains. On the other hand, when Naive Bayes is trained on our corpus, there are little variance in different domains (Table 5). This experiment indicates that our corpus is relatively robust against the change of the domain compared with small do-

main specific corpus. We think this is because our corpus is large and balanced. Since we cannot always get domain specific corpus in real application, this is the strength of our corpus.

Table 7: Cross domain evaluation.

		Training		
		Computer	Restaurant	Car
Test	Computer	—	0.701	0.773
	Restaurant	0.757	—	0.755
	Car	0.751	0.711	—

## 6 Related Works

### 6.1 Learning the polarity of words

There are some works that discuss learning the polarity of words instead of sentences.

Hatzivassiloglou and McKeown proposed a method of learning the polarity of adjectives from corpus (Hatzivassiloglou and McKeown, 1997). They hypothesized that if two adjectives are connected with conjunctions such as 'and/but', they have the same/opposite polarity. Based on this hypothesis, their method predicts the polarity of adjectives by using a small set of adjectives labeled with the polarity.

Other works rely on linguistic resources such as WordNet (Kamps et al., 2004; Hu and Liu, 2004; Esuli and Sebastiani, 2005; Takamura et al., 2005). For example, Kamps et al. used a graph where nodes correspond to words in the WordNet, and edges connect synonymous words in the WordNet. The polarity of an adjective is defined by its shortest paths from the node corresponding to 'good' and 'bad'.

Although those researches are closely related to our work, there is a striking difference. In those researches, the target is limited to the polarity of words and none of them discussed sentences. In addition, most of the works rely on external resources such as the WordNet, and cannot treat words that are not in the resources.

## 6.2 Learning subjective phrases

Some researchers examined the acquisition of subjective phrases. The subjective phrase is more general concept than opinion and includes both positive and negative expressions.

Wiebe learned subjective adjectives from a set of seed adjectives. The idea is to automatically identify the synonyms of the seed and to add them to the seed adjectives (Wiebe, 2000). Riloff et al. proposed a bootstrapping approach for learning subjective nouns (Riloff et al., 2003). Their method learns subjective nouns and extraction patterns in turn. First, given seed subjective nouns, the method learns patterns that can extract subjective nouns from corpus. And then, the patterns extract new subjective nouns from corpus, and they are added to the seed nouns. Although this work aims at learning only nouns, in the subsequent work, they also proposed a bootstrapping method that can deal with phrases (Riloff and Wiebe, 2003). Similarly, Wiebe also proposes a bootstrapping approach to create subjective and objective classifier (Wiebe and Riloff, 2005).

These works are different from ours in a sense that they did not discuss how to determine the polarity of subjective words or phrases.

## 6.3 Unsupervised sentiment classification

Turney proposed the unsupervised method for sentiment classification (Turney, 2002), and similar method is utilized by many other researchers (Yu and Hatzivassiloglou, 2003). The concept behind Turney's model is that positive/negative phrases co-occur with words like 'excellent/poor'. The co-occurrence statistic is measured by the result of search engine. Since his method relies on search engine, it is difficult to use rich linguistic information such as dependencies.

## 7 Conclusion

This paper proposed a fully automatic method of building polarity-tagged corpus from HTML documents. In the experiment, we could build a corpus consisting of 126,610 sentences.

As a future work, we intend to extract more opinion sentences by applying this method to larger HTML document sets and enhancing extraction rules. Another important direction is to investigate more precise model that can classify or extract opinions, and learn its parameters from our corpus.

## References

- Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the WWW*, pages 519–528.
- Andrea Esuli and Fabrizio Sebastiani. 2005. Determining the semantic orientation of terms through gloss classification. In *Proceedings of the CIKM*.
- Vasileios Hatzivassiloglou and Katherine R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the ACL*, pages 174–181.
- Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the KDD*, pages 168–177.
- Jaap Kamps, Maarten Marx, Robert J. Mokken, and Maarten de Rijke. 2004. Using wordnet to measure semantic orientations of adjectives. In *Proceedings of the LREC*.
- Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. 2002. Mining product reputations on the web. In *Proceedings of the KDD*.
- Bo Pang, Lillian Lee, and Shivakumar Vaihyathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the EMNLP*.
- Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the EMNLP*.
- Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the CoNLL*.
- Hiroya Takamura, Takashi Inui, and Manabu Okumura. 2005. Extracting semantic orientation of words using spin model. In *Proceedings of the ACL*, pages 133–140.
- Peter D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the ACL*, pages 417–424.
- Janyce Wiebe and Ellen Riloff. 2005. Creating subjective and objective sentence classifiers from unannotated texts. In *Proceedings of the CICLing*.
- Janyce M. Wiebe. 2000. Learning subjective adjectives from corpora. In *Proceedings of the AAAI*.
- Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the HLT/EMNLP*.
- Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the EMNLP*.