Extracting sentiment as a function of discourse structure and topicality *

Maite Taboada
Department of Linguistics
Simon Fraser University
Burnaby, B.C. V5A 1S6, Canada
mtaboada@sfu.ca

Kimberly Voll
Department of Computer Science
University of British Columbia
Vancouver, B.C. V6T 1Z4, Canada
kvoll@cs.ubc.ca

Julian Brooke
Department of Linguistics
Simon Fraser University
Burnaby, B.C. V5A 1S6, Canada
jab18@sfu.ca

Technical Report 2008-20
School of Computing Science, Simon Fraser University

December 2, 2008

Abstract

We present an approach to extracting sentiment from texts that makes use of contextual information. Using two different approaches, we extract the most relevant sentences of a text, and calculate semantic orientation weighing those more heavily. The first approach makes use of discourse structure via Rhetorical Structure Theory, and extracts nuclei as the relevant parts; the second approach uses a topic classifier built using support vector machines, which extracts topic sentences from texts. The use of weights on relevant sentences shows an improvement over word-based methods that consider the entire text equally. In the paper, we also describe an enhancement of our previous word-based methods in the treatment of intensifiers and negation, and the addition of other parts of speech beyond adjectives.

*This work was supported by the Natural Sciences and Engineering Research Council of Canada, under a Discovery Grant and a University Faculty Award to Maite Taboada. We also thank members of the Sentiment Research Group at SFU for their feedback, and in particular Vita Markman, Milan Tofiloski and Ping Yang for their help in ranking our dictionaries.

1 Introduction

Semantic orientation (SO) is a measure of subjectivity and opinion in text. It usually captures an evaluative factor (positive or negative) and potency (degree to which the document in question is positive or negative) towards a subject topic, person or idea (Osgood et al., 1957). When used in the analysis of public opinion, such as the automated interpretation of online product reviews, semantic orientation can be extremely helpful in marketing, measures of popularity and success, and compiling reviews.

There are two main approaches to the problem of extracting semantic orientation automatically. The "word counting", or semantic approach involves calculating orientation for a document from the semantic orientation of individual words or phrases, e.g., Turney (2002). The machine-learning approach uses n-grams as features to train a text classifier, e.g., Pang et al. (2002).

Most of the semantic research has focused on adjectives. A list of adjectives and corresponding SO values are compiled into a dictionary. For a given text, all adjectives are extracted and assigned SO based on this dictionary. The SO scores are in turn aggregated to arrive at a single score for that text.

On reading any document, it becomes apparent that aspects of the local context of an adjective need to be taken into account in SO assessment, such as negation (e.g., *not good*) and intensification (e.g., *very good*), proposed by Polanyi and Zaenen (2006). Recent research by Kennedy and Inkpen (2006) has concentrated on implementing those insights. Further to this, we postulate that not only the local context of the adjective, but also the global context of the entire text, play a role in determining semantic orientation. The existing semantic model is well placed to take advantage of research into related areas that explore context, such as contextual valence shifters (Polanyi and Zaenen, 2006), discourse parsing (Soricut and Marcu, 2003) and extraction of subjective sentences (Wiebe et al., 2004).

In this paper, we present a proposal for combining local and global information in the determination of semantic orientation. We first extract sentiment-bearing words (including adjectives, verbs, nouns and adverbs), and use them to calculate semantic orientation, taking into account valence shifters (intensifiers, downtoners and negation). We then motivate and present two new methods to calculate SO using the most relevant sentences in the text. In the first method, we parse the structure of the text and extract the main parts (*nuclei*, as defined within Rhetorical Structure Theory), in order to calculate semantic orientation by weighing the nuclei more heavily. In the second method, we extract topic sentences, and place a higher weight on the words found within them. We compare the success of these methods, and conclude that significant gains can be made when we consider both local and global context in the calculation of semantic orientation.

2 SO-CAL, the Semantic Orientation CALculator

Much of the previous research in extracting semantic orientation has focused on adjectives as the primary source of subjective content in a document (Hatzivassiloglou and McKeown, 1997; Taboada et al., 2006; Turney, 2002; Turney and Littman, 2003). In general, the SO of an entire document is the combined effect of the adjectives found within, based upon a

dictionary of adjective rankings (scores). The dictionary can be created in different ways: manually, using existing dictionaries such as the General Inquirer (Stone et al., 1966), or semi-automatically, making use of resources like WordNet (Esuli and Sebastiani, 2006; Fellbaum, 1998). More frequently, the dictionary is produced automatically via association, where the score for each new adjective is calculated using the frequency of the proximity of that adjective with respect to one or more seed words. "Seed words" refers to a small set of words with strong negative, or positive, associations, such as *excellent* or *abysmal*. In principle, a positive adjective should occur more frequently alongside the positive seed words, and thus will obtain a positive score, while negative adjectives will occur most often alongside negative seed words, thus obtaining a negative score. The association is usually calculated following Turney's method for computing mutual information (Turney, 2002; Turney and Littman, 2003).

To determine the overall SO score of a document, we use our SO-CAL (Semantic Orientation CALculator) software. Our previous version of SO-CAL relied on an adjective dictionary to predict the overall SO of a document, using a simple aggregate-and-average method: The individual scores for each adjective in a document are added together, and then divided by the total number of adjectives in that document. As we describe below, the current version of SO-CAL takes other parts of speech into account.

It is important to note that how a dictionary is created affects the overall accuracy of subsequent results. Taboada et al. (2006), for example, report on experiments using different search engines and operators. We followed Turney's basic approach to calculate orientation, experimenting with the context for the web-based search. Turney's method uses the number of hits (i.e. web documents) in which a word occurs in the vicinity of positive and negative seed words. Pointwise Mutual Information is then applied to produce a semantic orientation score for the word in question. Turney's original proposal used the Altavista search engine, which provided a NEAR operator: The target word was considered only if it appeared within ten words to the right or left of the seed word. Altavista has since discontinued the NEAR operator, however, thus we switched to using the Google API and the operator AND. The disadvantage of AND is that a hit is computed as long as target and seed word appear anywhere within the same document, where that document might be several pages long. As reported in Taboada et al. (2006), the resulting dictionary was still usable, though less accurate. Our main concern with the Google API, however, was its instability. When rerun, the results for each word were subject to change, sometimes by extreme amounts, something that Kilgarriff (2007) also notes, arguing against the use of Google for linguistic research.

In light of this, an additional dictionary was produced by hand-tagging all adjectives on a scale ranging from -5 for extremely negative, to $+5$ for extremely positive, where 0 indicates a neutral word. Although clearly not as scaleable, and subject to risk of bias, this gave us a solid dictionary for testing our adjective analyses and a point of comparison for evaluating the utility of the Google-generated dictionaries.

The original version of the dictionary contained 3,306 adjectives, extracted from our corpus. The automatic extraction of scores from the web using Pointwise Mutual Information provides values such as those shown in Table 1. The table also provides the hand-tagged values for those adjectives. Note that the automated extraction from the corpus allows the generation of a score for an adjective such as *unlistenable*, unlikely to be present in a human-generated list.

Table 1: Automatically-generated versus manual scores for some sample adjectives

Word	**Automatic**	**Hand-ranked**
air-conditioned	9.11	3
configurable	3.61	2
flawless	2.03	5
ghastly	−6.84	−5
listenable	−0.87	2
stand-offish	−4.85	−2
rude	−4.62	−3
tedious	−0.88	−3
top-quality	5.33	5
unlistenable	−7.84	−5

We use the automatically-created dictionary as the first step in our experiments with methods for detecting sentiment. To run our experiments, we use the corpus described in Taboada and Grieve (2004) and Taboada et al. (2006), which consists of a collection of Epinions reviews[1] extracted on eight different categories: books, cars, computers, cookware, hotels, movies, music, and phones. Within each collection, the reviews were split into 25 positive and 25 negative reviews, for a total of 50 in each category, and a grand total of 400 reviews in the corpus (279,761 words). We determined whether a review was positive or negative through the "recommended" or "not recommended" feature provided by the review's author.

In addition, we added a new set of 50 movie reviews as a comparison point. These reviews (Movies2), also split into positive and negative, came from Bo Pang's data[2], a large dataset of 2,000 texts (Pang and Lee, 2004, 2005; Pang et al., 2002). Since most of our development had been done on the 400 Epinions reviews, we use the extra movie dataset as a comparison benchmark.

The first comparison was performed using dictionaries consisting exclusively of adjectives, one Google-generated and the other one hand-ranked. The hand-ranked dictionary was built by one of the authors of this paper, augmented by another one of the authors, and checked for consistency by three more researchers. Some of the adjectives in our automatically generated dictionary were judged to have no semantic orientation, so we trimmed our hand-ranked dictionary to only those 1,982 words which expressed sentiment, and, for comparison, created a version of the Google dictionary with just those words. We ran the 400 Epinions reviews using each version of the dictionary. Results are presented in Table 2[3]. Based on these results, we chose to use the hand-ranked adjective dictionary for all subsequent testing.

[1] http://www.epinions.com

[2] Available from: http://www.cs.cornell.edu/People/pabo/movie-review-data/.

[3] The difference between the full Google dictionary and the hand-ranked is significant: $\chi^2_{df=1} = 5.79, p < 0.05$

Table 2: Google vs. hand-ranked dictionary

Dictionary	Word Count	Performance
Google (full)	3306	53.75%
Google	1982	56.00%
Hand-ranked	1982	59.75%

Table 3: Examples from the noun and verb dictionaries

Word	SO Value
hate (verb)	−4
hate (noun)	−4
inspire	2
inspiration	2
masterpiece	4
fabricate	−2
sham	−3
delay	−1
relish	4
determination	1

2.1 Nouns, verbs, and adverbs

In the following example, adapted from Polanyi and Zaenen (2006), we see that other lexical items can carry important semantic polarity information.

(1)

a) The young man strolled$^+$ purposefully$^+$ through his neighborhood$^+$.

b) The teenaged male strutted$^-$ cockily$^-$ through his turf$^-$.

Though the sentences have comparable literal meanings, the plus-marked nouns, verbs, and adverbs in (1a) indicate the positive orientation of the speaker towards the situation, whereas the minus-marked words in (1b) have the opposite effect.

In order to make use of this additional information, we created separate noun, verb, and adverb dictionaries, hand-ranked using the same 5 to −5 scale as our adjective dictionary. The noun dictionary contains 1,068 words, the verb dictionary 701 words, and the adverb dictionary 587 words. The nouns and verbs were mostly taken from the General Inquirer dictionary (Stone, 1997; Stone et al., 1966)[4], and supplemented by words appearing in our corpus. Those two dictionaries were created simultaneously, so that consistency was maintained among the various parts of speech. A few examples are shown in Table 3.

One difficultly with nouns and verbs is that they often have both neutral and non-neutral connotations. In the case of *inspire* (or *determination*), there is a very positive meaning (example 2) as well as a rather neutral meaning (example 3).

[4]Available on-line (http://www.wjh.harvard.edu/∼inquirer/).

Table 4: Examples from the adverb dictionary

Word	SO Value
excruciatingly	−5
inexcusably	−3
foolishly	−2
satisfactorily	1
purposefully	2
hilariously	4

(2) The teacher inspired her students to pursue their dreams.

(3) This movie was inspired by true events.

Except when one sense was very uncommon, the value chosen reflected an averaging across possible interpretations. All nouns and verbs encountered in the text were stemmed, and the form (singular or plural, past tense or present tense) was not taken into account in the calculation of SO value. As with the adjectives, there were a great deal more negative words than positive ones.

Except for a small selection of words that were added or modified by hand, our adverb dictionary was built automatically using our adjective dictionary. When our calculator came across something tagged as an adverb that was not already in its dictionary, it would stem the word and try to match it to an adjective in the main dictionary. This works quite well for most adverbs (see examples in Table 4).

In other cases, e.g., *essentially*, there is a mismatch between the meaning (or usage pattern) of the adverb as compared the adjective it is based on, and the value must be manually corrected.

2.2 Intensification

Quirk et al. (1985) classify intensifiers into two major categories, depending on their polarity: amplifiers (e.g., *very*) increase the semantic intensity of a neighbouring lexical item whereas downtoners (e.g., *slightly*) decrease it. Other researchers, (Kennedy and Inkpen, 2006; Polanyi and Zaenen, 2006) have implemented intensifiers using simple addition and subtraction, i.e., if a positive adjective has an SO value of 2, an amplified adjective would have an SO value of 3, and a downtoned adjective an SO value of 1. One problem with this kind of approach is that it does not account for the wide range of intensifiers within the same subcategory. *Extraordinarily*, for instance, is a much stronger amplifier than *rather*. Another concern is that the amplification of already "loud" items should involve a greater overall increase in intensity when compared to more subdued counterparts (compare *truly fantastic* with *truly okay*); in short, intensification should also depend on the item being intensified.[5] In our system, intensification is modelled using modifiers, with each intensifying

[5]Martin and White (2005) also suggest that the effect is different according to the polarity of the item being intensified. We have not explored that possibility.

Table 5: Percentages for some intensifiers

Intensifier	**Modifier %**
somewhat	−30%
pretty	−10%
really	+15%
very	+25%
extraordinarily	+50%
(the) most	+100%

word having a percentage associated with it; amplifiers are positive, whereas downtoners are negative, as shown in Table 5.

For example, if *sleazy* has an SO value of −3, *somewhat sleazy* would have an SO value of: $-3+(3\times30\%) = -2$. If *excellent* has a SO value of 5, *the most excellent movie I've seen this year* would have an SO value of: $5 + (5 \times 100\%) = 10$. Intensifiers are additive: If *good* has a value of 3, then *really very good* has an SO value of: $3 + (3 \times 15\%) + (3 \times 25\%) = 4.3$.

Since our intensifiers are implemented using a percentage scale, they can fully reflect the variety of intensifying words as well as the SO value of the item being modified. This scale can be applied to other parts of speech, given that adjectives, adverbs, and verbs can all use the same set of intensifiers, as seen in example (4), where *really* modifies an adjective (*fantastic*), an adverb (*well*) and a verb (*enjoyed*).

(4)

a) The performances were all really fantastic.

b) Zion and Planet Asai from the Cali Agents flow really well over this.

c) I really enjoyed most of this film.

Nouns, however, are modified exclusively by adjectives. We are able to take into account some kinds of modification using our main adjective dictionary; however there are a small class of adjectives which would not necessarily amplify or downtone correctly if considered in isolation, as seen in the following (invented) examples. Here, adjectives such as *total* do not have a semantic orientation of their own, but like adverbial intensifiers they contribute to the interpretation of the word that follows it; *total failure* is presumably worse than just *failure*. Thus we have a separate dictionary for adjectival intensifiers. When an intensifying adjective appears next to an SO-valued noun, it is treated as an intensifier rather than as a separate SO-bearing unit.

(5)

a) The plot had huge problems.

b) They have made clear progress.

c) This is a total failure.

d) It's an insignificant criticism.

A small group of commonly appearing intensifiers have special semantic properties that need to be accounted for. Consider *too*: When it appears with negative adjectives, it has an obviously intensifying effect, but, outside of fixed expressions in colloquial speech (e.g., *too cool*), it acts somewhat like a negator. We handle this by simply assigning a fixed negative value to it.

(6)

a) It is too predictable. $(-2 + (2 \times 50\%) = -3)$

b) He was just too pretty for this kind of role. (-2)

c) He wrote the character Luke a little too intelligent for his age. $(-2 - (-2 \times 50\%) = -1)$

In the last example, *too* changes the base SO value of *too intelligent* to -2, then *a little* downtones that by 50% to -1.

2.3 Negation

The obvious approach to negation is simply to reverse the polarity of the lexical item next to a negator, changing *good* $(+3)$ into *not good* (-3), for example. However, there are a number of subtleties related to negation that need to be taken into account. One is the fact that there are a number of negators, including *not*, *none*, *nobody*, *never*, and *nothing*, and other words, such as *lack*, which have equivalent effect, some of which might occur at a significant distance from a lexical item that they affect; a backwards search is required to find these negators, one that is tailored to the particular part of speech involved. We assume that for adjectives and adverbs the negation is fairly local, though it is sometimes necessary to look past determiners, copulas, and certain verbs, as we see in example (7).

(7)

a) Nobody gives a good performance in this movie (*nobody* negates *good*).

b) Out of every one of the fourteen tracks, none of them approach being weak and are all stellar (*none* negates *weak*).

c) Just a V-5 engine, nothing spectacular (*nothing* negates *spectacular*).

The other parts of speech are more difficult to characterize, and we have chosen simply to search backward through the entire sentence until a negator is found. This allows us to capture the true effects of negation raising (Horn, 1989), where the negator for a verb moves up and attaches to the verb in the matrix clause. In the following examples the *don't* that negates the verb *think* is actually negating the embedded clauses.

(8) I don't wish to reveal much else about the plot because I don't think it is worth mentioning.

(9) Based on other reviews, I don't think this will be a problem for a typical household environment.

Another issue is whether a polarity flip is the best way to quantify negation. Though it seems to work well in certain cases, it fails miserably in others. Consider *excellent*, a +5 adjective: if we negate it, we get *not excellent*, which intuitively is a far cry from *atrocious*, a −5 adjective. In fact, *not excellent* seems more positive than *not good*, which would negate to a −3. In order to capture these pragmatic intuitions, we implemented another method of negation, a polarity shift. Instead of changing the sign, the SO value is shifted toward the opposite polarity by a fixed amount (in our current implementation, 4). Thus a +2 adjective is negated to a −2, but the negation of a −3 adjective (say, *sleazy*) is only slightly positive, an effect we could call "damning with faint praise." Below are a few examples from our corpus.

(10)

a) She's not terrific $(5 - 4 = 1)$ but not terrible $(-5 + 4 = -1)$ either.

b) Cruise is not great $(4 - 4 = 0)$, but I have to admit he's not bad $(-3 + 4 = 1)$ either.

c) This CD is not horrid $(-5 + 4 = -1)$.

In each case, the negation of a strongly positive or negative value reflects an ambivalence which is correctly captured in the shifted value. Further (invented) examples are presented in example (11).

(11)

a) Well, at least he's not sleazy. $(-3 \rightarrow 1)$

b) Well, it's not dreadful. $(-4 \rightarrow 0)$

c) It's just not acceptable. $(1 \rightarrow -3)$

d) It's not a spectacular film, but... $(5 \rightarrow 1)$

As in the last example, it is very difficult to negate a strongly positive word without implying that a less positive one is to some extent true, and thus our negator becomes a downtoner.

A related problem for the polarity flip model, as noted by Kennedy and Inkpen (2006), is that negative polarity items interact with intensifiers in undesirable ways. *Not very good*, for instance, comes out more negative than *not good*. Another way to handle this problem while preserving the notion of a polarity flip is to allow the negative item to flip the polarity of both the adjective and the intensifier; in this way, an amplifier becomes a downtoner:

- Not good $= (3 \times -1) = -3$
- Not very good $= ((-3 \times (25\%)) \times -1) + (3 \times -1) = -2.2$

Compare with the polarity shift version, which is only marginally negative:

- Not good $= 3 - 4 = -1$

- Not very good $= (3 + 3 \times (25\%)) - 4 = -0.2$

The problems with polarity shift could probably be resolved by fine-tuning SO values and modifiers; however the polarity flip model seems fundamentally flawed. Polarity shifts seem to better reflect the pragmatic reality of negation, and is supported by the work of Horn (1989), who points out that affirmative and negative sentences are not symmetrical.

Our calculator also handles modals, albeit in a very basic way; SO values of predicates are nullified when preceded by a a modal (e.g., *would*, *could*, *should*).

(12) This should have been a great movie. $(3 \rightarrow 0)$

One other interesting aspect of the pragmatics of negation is that negative statements tend to be perceived as more marked than their affirmative counterparts, both pragmatically and psychologically (Horn, 1989; Osgood and Richards, 1973), in terms of linguistic form as well as across languages (Greenberg, 1966), and in frequency distribution, with negatives being less frequent (Boucher and Osgood, 1969). People often use roundabout ways to express negation, which makes negative sentiment more difficult to identify in general.

A final remark with respect to our treatment of negation is that it is very narrow. We consider mostly negators, but not negative polarity items (NPIs). In some cases, searching for an NPI would be more effective than searching for a negator. NPIs occur in negative sentences, but also in nonveridical contexts (Giannakidou, 1998; Zwarts, 1995), which also affect semantic orientation. For instance, *any* occurs in contexts other than negative sentences, a shown in example (13), from Giannakidou (2001, p. 99), where in all cases the presence of *any* is due to a nonveridical situation. Using NPIs would allow us to reduce semantic orientation in such contexts.

(13)

a) Did you find any interesting books?

b) Pick any apple!

c) He might come any moment now.

d) I insist you allow anyone in.

2.4 Performance

The final versions of our dictionaries contain 2,132 adjectives (after adding more adjectives from the Bo Pang reviews), 1,068 nouns, 701 verbs, 587 adverbs, and 84 intensifiers (including both amplifiers and downtoners). Each dictionary also has associated with it a stop-word list. For instance, the adjective dictionary has a stop-word list that includes *more, much,* and *many*, which are tagged as adjectives by the Brill tagger.

When we tested our performance with the new verb, noun, and adjective dictionaries, we saw a significant improvement: 5.5% in our main Epinions corpus and 12% in the Movies 2 corpus (see Table 6).

It is worth remarking that nouns, verbs, and adverbs (without adjectives) seemed to do as well as, or better, than adjectives alone at predicting overall text sentiment. In fact,

Table 6: Comparison of performance using different dictionaries

Corpus	**Percent correct**		
	Adjs only	**Nouns, verbs, adverbs only**	**All word types**
Epinions	60.5%	66.5%	66.0%
Movies2	72.0%	74.0%	84.0%

Table 7: Comparison of performance using different features

SO-CAL Options	**Percent Correct**
Baseline (only adjectives)	61.78%
All words (nouns, verbs, adjs, advs)	67.1%
All words + negation (shift)	68.6%
All words neg (shift) + intensification	69.8%
All words + neg (shift) + int + modals	70.0%
All words + neg (switch) + int + mod	69.6%
All words + neg (shift) + int (x10) + mod	72.7%

certain types of reviews (for instance, computer reviews) do more than 10% better when these new dictionaries are used instead of the adjective dictionary. For the Epinions corpus, we actually see a drop in performance when adjectives are once again taken into account, though testing with Bo Pang's full data set has shown that this is atypical; we believe it is in part due to our adjective dictionaries being built using exactly those adjectives in the Epinions corpus. Full coverage may result in worse performance on reviews which involve, for instance, lengthy plot summaries or discussion of other products.

In order to test this theory, we created another 50-movie review set and automatically extracted all lexical items that were not in our dictionaries, manually removing those with no semantic orientation. We were left with 116 adjectives, 62 nouns, 43 verbs, and 7 adverbs. These words were given SO values and added to create alternate versions of the dictionaries. However, performance on the new reviews actually dropped 4% (from 70% to 66%) with addition of these new words, suggesting that accuracy is not necessarily a function of coverage, and that simply adding words to the dictionary will not lead to sustainable improvement; in fact, it might have the opposite effect.

The addition of other features had a less dramatic but still noticeably positive effect on overall performance. The results, averaged across our entire corpus (including Movies2), are summarized in Table 7 [6] .

As reported in Kennedy and Inkpen (2005), basic negation is more useful than basic intensification. However, we discovered that if the volume of the intensifiers is increased to a significant degree (here, 10 times normal, i.e., +50% becomes +500%), there is a marked boost in performance. This may reflect the fact that intensifiers are more likely to be used in expressions of subjective opinion, and less likely to be used in the kind of objective description that ends up as semantic noise. These sorts of patterns should obviously be taken into consideration, however probably not at the level of SO calculation for individual

[6] The improvement of All words + neg (shfit) + int + modals, i.e., the 70.0: $\chi^2_{df=1} = 12.88, p < 0.001$

words; a broader approach is needed to determine exactly which parts of the document are relevant to the calculation.

There is another intriguing explanation for this increase: It is the result of increasing the volume of downplayers such that they actually become negators (e.g., a −20% becomes a −200%, which is equivalent to a flip in polarity). In texts where there is some ambivalence towards the subject, the grudging approval or disapproval implicit in a downplayed SO-carrying word may actually signal that the true orientation of the author is in the other direction; consider example (14), from the beginning of a negative review.

(14) To begin with, I only mildly like Will Farrell.

A module which could detect and ignore concessionary clauses in a review would likely improve performance on the polarity recognition task. However, if the SO value of the text is ultimately intended to reflect not only the polarity of sentiment but also the degree, downplayed words should be neither negated nor discarded insofar as they indicate on-topic opinion.

Though the difference is small, we see here that shifted polarity negation does, on average, perform better than switched polarity negation. Another interesting result (Table 8) is that our performance on positive reviews is substantially better than negative reviews (run with all options and shifted negation). This is despite the fact that all of our dictionaries contain far more negative words than positive ones. As noted, people often avoid negation and negative terms even when expressing negative opinions, making the detection of text sentiment difficult for systems which depend solely on these indicators. Table 8 shows the performance of the SO-CAL system across different review types, and in positive and negative texts. In order to arrive at these results, we simply compared the output of SO-CAL to the "recommended" or "not recommended" field of the reviews. An output above zero is considered positive (recommended), and negative if below zero. The overall performance is decent (70%), but the breakdown shows a very weak performance on negative reviews. As it has already been pointed out with regard to reviews (Dave et al., 2003; Turney, 2002), negative reviews are notoriously difficult to analyze because they do not necessarily contain negative words. However, our performance is better on art-related reviews (books, music and movies) than in consumer products. We hypothesize this is because consumer product reviews contain more factual information that the reader is required to interpret as positive or negative (for instance, the range for a cordless phone or the leg room in the back seat of a car).

Finally, we ran the same system on the full 2,000 reviews provided by Bo Pang. The result is an accuracy of 56.10% on negative reviews, and 87% for positive ones, with an average of 71.55%. Although the results are below machine learning methods, they are above the human baseline proposed by Pang et al. (2002), reported to be between 50 and 69%.

3 Extracting relevant sentences

After extensive experimentation with different approaches to keyword-based sentiment extraction of the type shown in the previous section, we are convinced that we need to move on to consider contextual information. One could continue to change parameters, develop

Table 8: Performance across review types and on positive and negative reviews

Sub-corpus	**Percent correct**		
	Positive	**Negative**	**Overall**
Books	84.0%	56.0%	70.0%
Cars	100.0%	32.0%	66.0%
Computers	100.0%	48.0%	74.0%
Cookware	100.0%	20.0%	60.0%
Hotels	100.0%	16.0%	58.0%
Movies	84.0%	52.0%	68.0%
Movies2	84.0%	92.0%	88.0%
Music	96.0%	52.0%	74.0%
Phones	100.0%	44.0%	72.0%
Total	94.2%	45.8%	70.0%

more sophisticated methods to deal with negation, and address multiple issues with intensification. Our belief is that this would only result in small increases in performance, and would not address the main issue, namely that large amounts of noise are included along with the relevant information.

It is readily apparent to an individual reading a review text that some parts are more relevant than others to the overall sentiment expressed: Some sentences are on-topic, whereas others are off-topic; some sentences are central to the intention of the writer, whereas others perform a supporting role. Taboada and Grieve (2004) previously found that assigning weights to adjectives based upon their placement in the text increased the performance of automatic SO extraction. In this section, we describe how we have attempted to extract relevant sentences using existing methods. The first method extracts nuclei within sentences using a discourse parser. The second method extracts sentences that are considered on-topic. In both cases, our SO-CAL system is then run on those sentences that were considered more relevant, with improved results as we shall see in the next subsections.

3.1 Discourse structure using SPADE

Our first approach consists of extracting rhetorical structure from the texts, assigning parts of the text to nucleus or satellite status, and then performing semantic orientation calculations only on the nuclei, i.e., the most important parts.

Rhetorical structure refers to the underlying relations between parts of a text, as postulated by Rhetorical Structure Theory (RST). RST characterizes text coherence through text relations (Mann and Thompson, 1988; Taboada and Mann, 2006), which represent functional correspondences between consecutively placed text spans and capture the intention behind the text. In analyzing the discourse structure of a text, it is possible to not only identify these spans, but also to determine which span is more central to the author's intention. Central spans are marked as nuclei, while less central, or supporting spans, are marked as satellites. As the dominant elements in a text, we hypothesize that the adjectives within the nuclei of a document are also more central to the overall sentiment, while avoiding potential

interference by the satellite adjectives, whose sentiment is arguably more tangential, or even irrelevant, to the text's overall sentiment.

Although there are many examples of manual annotation of rhetorical relations, automatic parsing of relations (discourse parsing) is far from an achievable goal. One partial discourse parser of relatively high success is Marcu and Soricut's SPADE parser (Soricut and Marcu, 2003), which parses the relationships within a sentence (although it does not address cross-sentential relationships). SPADE was trained on Wall Street Journal articles, and is therefore not tailored to the types of texts that we are considering. Despite its limitations, we used SPADE in our analysis as an experiment to show whether this avenue was worth exploring further. Each review in our test set is prepared and run through SPADE, creating a series of discourse-annotated files, each divided into nuclei and satellites. To calculate SO, the outer-most nuclei of each sentence are tagged as nuclei, and the rest as satellites. If a sentence consists of a single span, that span is also tagged as a nucleus. We then run this newly-tagged text file through SO-CAL to determine its SO value. As we explain in Section 4, we experimented with different ways of assigning weights to nuclei and satellites in SO-CAL.

3.2 Topic sentences

The second means by which we can measure relevance is topicality. Sentiment-bearing words may be tangential, or even irrelevant, such as the words *soggy* and *stale* in the following observation: *This movie was fantastic, although the popcorn was soggy and stale.* Here the sentiment is clearly positive toward the main topic, the movie, although the popcorn is mentioned negatively. Such sentences and collections of sentences (where the reviewer may discuss a tangential topic for a span of more than one sentence before returning to the main topic) indicate that not all words are relevant to the overall topic. For example, in reading a movie review for a particular movie, the reviewer may briefly depart from the movie in question to discuss another movie in which they saw the main actor. Their opinion about the second movie would not be relevant to their opinion of the movie for which the review was written.

Thus, if a word is found in an off-topic sentence, its score should not be counted in the overall SO, or it should at least be given a lower weight. That is, we require a topic classifier to apply to novel documents at the sentence level in order to determine which sentences are relevant and should be included in the sentiment analysis. To obtain this, we take inspiration from the work of Wilson et al. (2005). Using the WEKA software suite (Witten and Frank, 2005), we train a decision tree on the basis of on-topic/off-topic documents, using the ID3 algorithm. We use the resulting model to classify the individual sentences of the on-topic documents as on- or off-topic.

Our data is split into eight cases, where each Epinions topic is in turn used as the positive, or on-topic, instances for the classifier, while all other topics indicate negative, or off-topic, instances. The remaining instance attributes are determined on the basis of the words present within each document. Unfortunately, using all possible words (even if confined to only those present within the training set) creates an impossibly large attribute set.[7] Having

[7] In this particular experiment, this amounted to over 15,000 features on which to train.

too many features during training causes significant amounts of noise, leading to data overfit and consequently useless results. Thus, it was necessary to prune the set of attributes: Each word found in our corpus is listed in order of its occurrence rate. After various experiments with attribute set size, the top 500 most common words[8] were extracted to form our working attribute set[9]. Once created, the attribute list is used to generate a feature vector for each document, where a "1" indicates the presence of one or more occurrences of a word within a document, while a "0" indicates its absence. In addition, an attribute indicating on- or off-topic, for each document is also included in the vector. These vectors are then run through the classifier, training a total of eight models (one for each topic designation), each of which demonstrate a minimum 95-percent accuracy when tested using 10-fold cross validation.

Table 9: Accuracy of the topic classifier on whole texts

Sub-corpus	**Percent correct**
Books	98.4%
Cars	95.6%
Computers	96.4%
Cookware	98.0%
Hotels	97.3%
Movies	95.6%
Movies2	98.7%
Music	98.7%
Phones	96.2%

Since the ultimate goal is to determine the topicality of the individual sentences, not the entire document, the test set for each classifier model is formed from the individual sentences in each on-topic document (depending on the relevant topic/model). Each sentence results in a feature vector, generated in the same fashion as for the entire document, while the topic attribute is set to "unknown". After training, the on-topic sentences are marked as such. The new file is then run through SO-CAL to determine its SO value, weighing the topic/off-topic sentences differently. Note that this approach is unorthodox: We train using the 50 texts for each document set (e.g., movies, books, cars), but then we apply the topic classifier to detect topic sentences *within* a document.

4 Results

As we describe in the previous section, we run two methods of extracting relevant sentences: discourse parsing using SPADE and topic detection using WEKA. The next question to ask is how to weight the relevant versus the non-relevant sentences. One approach is a 1-0 weighting scheme, where only the relevant sentences are included in the SO-CAL analysis

[8] A stoplist of 300 words was used to remove the effect of the most frequently occurring, low-information-content words. This included additional words that were deemed irrelevant to the overall task.

[9] As a useful side effect, this also eliminated noise in the form of spelling errors, which arise as rare or single-use words.

(these results are provided below). However, we know that neither the SPADE extraction nor the topic detection are perfectly accurate, resulting in both precision and recall errors. Therefore, we established the following weighting scheme: 1.5 for relevant sentences; 0.5 for non-relevant (in either version of "relevance", nuclei and topics). We also combined the results of both methods for extracting relevant sentences, and weighed them: 2 if the sentence was relevant according to both SPADE and WEKA; 1.5 if the sentence was relevant in either method; 0.5 for all other sentences. As a final point of comparison, we also ran an experiment whereby only those sentences deemed to be *irrelevant* by both WEKA and SPADE were considered (i.e. such sentences were weighted 1, while all other sentences were weighted 0). Pang and Lee (2004) report a similar experiment.

In addition, and to counterbalance the positive bias that we already discussed in Section 2.4, we shifted the 0 point, and used a different value for the breakdown into positive and negatives. We calculated this new breaking point by finding the average review values for the positive and negative reviews, respectively, and calculating the difference between these two averages, which was 0.62. Thus, a review above 0.62 was positive, and below that value, negative. A similar normalization factor was used in Voll and Taboada (2007). The results of all our tests are found in Table 10.

Table 10: Overall performance of SPADE and WEKA with weights and normalization factor

	Normalization factor	
	0	**0.62**
SO-CAL	70.00%	72.00%
SPADE (1,0)	68.44%	73.78%
SPADE (1.5, 0.5)	71.56%	80.00%
WEKA (1,0)	65.24%	72.41%
WEKA (1.5, 0.5)	70.00%	80.67%
SPADE-WEKA (2, 1.5, 0.5)	70.67%	78.44%
SPADE-WEKA (*inverse:* 0, 1)	65.11%	52.89%

As the table shows, the normalization factor improves the performance of all methods (with the exception, naturally, of the last method, which is expected to deteriorate).[10] The first result to note is that a 1.5-0.5 weight is better than a "relevant only" approach, under both methods (SPADE and WEKA). This difference is statistically signicant both for SPADE ($\chi^2_{df=1} = 14.29, p < 0.001$) and for WEKA ($\chi^2_{df=1} = 16.77, p < 0.001$). One explanation is that sentences deemed to be irrelevant do, in fact, contribute to the overall sentiment of the text. In positive reviews, people are more likely to be charitable even when describing secondary aspects, and likewise, less so in negative reviews. Another explanation is that the methods for extracting relevance are fallible, and when adding some weight to the "irrelevant" sentences, we correct for this error.

Finally, the combined SPADE-WEKA data decreases the performance, but only slightly (and it is still significantly better than the baseline SO-CAL ($\chi^2_{df=1} = 9.27, p < 0.05$). We

[10] The WEKA 1-0 numbers exclude files that had no topic sentences after running SO-CAL, a total of 106 texts (24%).

believe this is also because the 1.5 weight for both SPADE and WEKA individually brings back non-sentiment-bearing topic sentences (as with the WEKA-only data).

It is also interesting to note that the inverse test, using only *irrelevant* sentences, supports the normalization factor. It shows that there is a positive bias of similar degree: The normalized result brings performance back to a 50% baseline.

As with the basic SO-CAL, the performance of both SPADE and WEKA varies across review types. The individual results of SO-CAL on the sentences extracted using SPADE are shown in Table 11. The system still performs better on positive reviews, even after normalization. It has a positive bias in cookware, hotels and phones. The exceptions are books and movies, and especially the second movie review set. The difference between books and movies versus other products is that the former contain plot descriptions that do not reflect user sentiment.

Table 11: Performance of SO-CAL with heavier weight on nuclei (1.5) and satellites (0.5), and break at 0.62

Sub-corpus	**Percent correct**		
	Positive	**Negative**	**Overall**
Books	60%	76%	68%
Cars	92%	76%	84%
Computers	92%	80%	86%
Cookware	100%	52%	76%
Hotels	92%	64%	78%
Movies	80%	84%	82%
Movies2	56%	96%	76%
Music	96%	88%	92%
Phones	96%	60%	78%
Total	84.89%	75.11%	80%

The next table shows the breakdown by review type on topic sentences, again with the best performing method (weights of 1.5-0.5, and normalization factor). As with SPADE, the performance is better on positive reviews in general.

One of the issues in the WEKA-based analysis is sentence length (and by extension, document length). Since the analysis relies on the presence of features within a sentence, a sentence that is too short may not have adequate (or any) topic features, thus forcing a false zero value. That is, the sentence may in fact be considered on topic, but fails to contain any of the words deemed salient by the topic classifier. In turn, a sentence may be on topic, yet contain no words of sentiment. In Table 12, we correct for these issues by applying a sentiment-word threshold to the WEKA-based analysis. Texts that contained no sentiment words after extracting topic sentences are omitted from the analysis. On its own, this is a hard problem to overcome given our current method of topic analysis; however, when combined with the SPADE analysis, the two measures of relevance compliment each other, resulting in a higher overall performance (without need for a sentiment-word threshold).

Table 12: Performance of SO-CAL with heavier weight on topic sentences (1.5), and break at 0.62

Sub-corpus	**Percent correct**		
	Positive	**Negative**	**Overall**
Books	60%	84%	72%
Cars	100%	72%	86%
Computers	96%	84%	90%
Cookware	100%	52%	76%
Hotels	92%	60%	76%
Movies	76%	88%	82%
Movies2	56%	96%	76%
Music	96%	84%	90%
Phones	96%	60%	78%
Total	85.78%	75.56%	80.67%

5 Comparison to other work

The SO-CAL improvements described in this paper have been directly inspired by the work of Polanyi and Zaenen (2006), who proposed that "valence shifters" change the base value of a word. We have implemented their idea in the form of intensifiers and downtoners, adding a treatment of negation that does not involve switching polarity, but instead shifting the value of a word when in the scope of a negator. Kennedy and Inkpen (2006) also implemented the ideas of Polanyi and Zaenen, seeing an improvement over basic positive and negative words using a variety of dictionaries, and different configurations of valence shifters. Kennedy and Inkpen's basic method starts at a performance level of 61.1%, and their best-performing method, adapted from Taboada and Grieve (2004), tops at 67.8%. Our current instantiation of the basic SO-CAL performs better, with results that range, depending on the configuration, from 67.1% to 72.7%, and with a top score when using relevant sentences of 80.67%.[11]

In addition to improving SO-CAL, we have shown how we can extract relevant sentences for a more accurate calculation of sentiment, also a suggestion of Polanyi and Zaenen (2006), who propose discarding the value of words in the *although* part of a concessive sentence. The most closely related work in this aspect is that of Wiebe and Riloff (2005); Wiebe et al. (2004); Wilson et al. (2006), who classify sentences as subjective and objective. However, they do not calculate polarity on the subjective sentences, and thus we cannot compare their work to ours[12]. A possible extension of our work is to use Wiebe et al.'s classifiers to extract subjective sentences, and calculate sentiment on those. In a sense, theirs is another method of finding the relevant portions of a text.

Pang and Lee (2004) extract subjective sentences from a text using a Naïve Bayes classifier, and use those to classify texts as positive or negative, achieving a statistically significant

[11] Kennedy and Inkpen also include a machine learning method, and a combination of keywords and machine learning, with a significant improvement over the keyword-based method. Their top result, with a combined method, is 86.2%.

[12] Wilson et al. (2005) extract sentiment, but only from phrases, not texts.

improvement over the sentiment calculator for the entire text (82.8% for the entire text; 86.4% for the extracted subjective parts). It is worth mentioning that they found differences across classifiers: The difference between support vector machine classifiers with or without subjectivity filtering was small. This may be relevant for us, since we believe that our topic classifier stands to improve.

6 Conclusions and future research

We have presented a word-based method for extracting sentiment from texts. Building on previous research that made use of adjectives, we extend our Semantic Orientation CALculator (SO-CAL) to other parts of speech. We also introduce intensifiers, and refine our approach to negation. The current results represent a statistically significant improvement over previous instantiations of the system.

We have shown that further improvements in word-based methods for sentiment detection need to come from analysis of the most relevant parts in a text. It is possible that small improvements in our dictionary will give rise to corresponding small improvements in results. However, we believe that further progress can only be made if we are able to identify the portions of the text that contain the most relevant expressions of sentiment.

Using SPADE's classification of sentences into nuclei and satellites (more and less important parts of the text), and a WEKA-built topic classifier, we apply the SO-CAL algorithm to relevant sentences. The results show that either method outperforms basic SO-CAL by a significant margin. In addition, we show improvement over previous work. In Voll and Taboada (2007), preliminary experiments using SPADE demonstrated a 69% performance level. Our higher baseline SPADE performance is a result of our improvements to SO-CAL.

The two methods to extract relevant sentences that we have implemented here can be further refined. Topic classification is certainly a well-known area, and better topic classifiers exist. Although most methods apply to documents, and not sentences within a document, sentence-based topic classification methods have been researched (Hovy and Lin, 1997). A similar approach would be to apply extractive text summarization, where the most important sentences in a document are extracted to produce a summary (Radev et al., 2004); (Teufel et al., 1999). In our case, we could produce the summary, and then perform sentiment orientation calculations on the sentences in the summary.

The avenue that we are most interested in pursuing, however, is the discourse-parsing one. The method for discourse parsing that we have used in this paper is quite limited. It builds discourse trees for structures within the sentence only, and it was trained on newspaper articles. It is no surprise, then, that it does not perform very well on our data. A more robust discourse parser, even if it only parses at the sentence level, would improve our results. Furthermore, we would like to explore other methods for calculating sentiment out of discourse trees. Here, we have used only nuclei, regardless of the type of relation between nucleus and satellite. For instance, in a *Summary* relation, we would be interested mostly in the nucleus. Similarly for *Elaboration* and *Concession* relations. A *Condition* relation, on the other hand, may warrant a different approach. Consider the following example from our corpus. A correct parse would have assigned satellite status to the first clause in the sentence, whereas the second clause would be a nucleus. Disregarding the satellite means

that we miss the condition imposed on *perfectly*. In this case, the aggregation of *Condition* ought to take into account the satellite as well as the nucleus.

(15) If the plot had been more gripping, more intense, [N] this would have worked perfectly.

Our current work is focused on developing discourse parsing methods, both general and specific to the review genre. At the same time, we will investigate different aggregation strategies for the different types of relations in the text.

References

Boucher, J. D. and C. E. Osgood (1969). The pollyanna hypothesis. *Journal of Verbal Learning and Verbal Behaviour 8*, 1–8.

Dave, K., S. Lawrence, and D. M. Pennock (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the Twelfth International World Wide Web Conference (WWW 2003)*, Budapest, Hungary.

Esuli, A. and F. Sebastiani (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy, pp. 417–422.

Fellbaum, C. (Ed.) (1998). *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Giannakidou, A. (1998). *Polarity Sensitivity as (Non)Veridical Dependency*. Amsterdam and Philadelphia: John Benjamins.

Giannakidou, A. (2001). Varieties of polarity items and the (non)veridicality hypothesis. In J. Hoeksema, H. Rullmann, V. SǴnchez-Valencia, and T. van der Wouden (Eds.), *Perspectives on Negation and Polarity Items*, pp. 99–127. Amsterdam and Philadelphia: John Benjamins.

Greenberg, J. H. (1966). *Language Universals, with Special Reference to Feature Hierarchies*. The Hague: Mouton.

Hatzivassiloglou, V. and K. McKeown (1997). Predicting the semantic orientation of adjectives. In *Proceedings of 35th Meeting of the Association for Computational Linguistics*, Madrid, Spain, pp. 174–181.

Horn, L. R. (1989). *A Natural History of Negation*. Chicago: University of Chicago Press.

Hovy, E. and C. Y. Lin (1997). Automated text summarization in summarist. In *Proceedings of ACL/EACL Workshop on Intelligent Scalable Text Summarization*, Madrid, pp. 18–24.

Kennedy, A. and D. Inkpen (2006). Sentiment classification of movie and product reviews using contextual valence shifters. *Computational Intelligence 22*(2), 110–125. Citation of Taboada and Grieve. They also used our list of adjectives.

Kilgarriff, A. (2007). Googleology is bad science. *Computational Linguistics 33*(1), 147–151.

Mann, W. C. and S. A. Thompson (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text 8*(3), 243–281.

Martin, J. R. and P. White (2005). *The Language of Evaluation*. New York: Palgrave. Citation of Taboada and Grieve.

Osgood, C. E. and M. M. Richards (1973). From yang and yin to and or but. *Language 49*(2), 380–412.

Osgood, C. E., G. Suci, and P. Tannenbaum (1957). *The Measurement of Meaning*. Urbana: University of Illinois.

Pang, B. and L. Lee (2004). A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of 42nd Meeting of the Association for Computational Linguistics*, Barcelona, Spain, pp. 271–278.

Pang, B. and L. Lee (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of ACL 2005*, Ann Arbor, MI.

Pang, B., L. Lee, and S. Vaithyanathan (2002). Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of Conference on Empirical Methods in NLP*, pp. 79–86.

Polanyi, L. and A. Zaenen (2006). Contextual valence shifters. In J. G. Shanahan, Y. Qu, and J. Wiebe (Eds.), *Computing Attitude and Affect in Text : Theory and Applications*, pp. 1–10. Dordrecht: Springer.

Quirk, R., S. Greenbaum, G. Leech, and J. Svartvik (1985). *A Comprehensive Grammar of the English Language*. London: Longman.

Radev, D., H. Jing, M. Stys, and D. Tam (2004). Centroid-based summarization of multiple documents. *Information Processing and Management 40*, 919–938.

Soricut, R. and D. Marcu (2003). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of Human Language Technology and North American Association for Computational Linguistics Conference (HLT-NAACL'03)*, Edmonton, Canada.

Stone, P. J. (1997). Thematic text analysis: New agendas for analyzing text content. In C. Roberts (Ed.), *Text Analysis for the Social Sciences*. Mahwah, NJ: Lawrence Erlbaum.

Stone, P. J., D. C. Dunphy, M. S. Smith, and D. M. Ogilvie (1966). *The General Inquirer: A Computer Approach to Content Analysis*. Cambridge, MA: MIT Press.

Taboada, M., C. Anthony, and K. Voll (2006). Creating semantic orientation dictionaries. In *Proceedings of 5th International Conference on Language Resources and Evaluation (LREC)*, Genoa, Italy, pp. 427–432.

Taboada, M. and J. Grieve (2004). Analyzing appraisal automatically. In *Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text (AAAI Technical Report SS-04-07)*, Stanford University, CA, pp. 158–161. AAAI Press.

Taboada, M. and W. C. Mann (2006). Rhetorical structure theory: Looking back and moving ahead. *Discourse Studies 8*(3), 423–459.

Teufel, S., J. Carletta, and M. Moens (1999). An annotation scheme for discourse-level argumentation in research articles. In *Proceedings of 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL'99)*, Bergen, Norway, pp. 110–117.

Turney, P. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of 40th Meeting of the Association for Computational Linguistics*, pp. 417–424.

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

Voll, K. and M. Taboada (2007). Not all words are created equal: Extracting semantic orientation as a function of adjective relevance. In *Proceedings of the 20th Australian Joint Conference on Artificial Intelligence*, Gold Coast, Australia, pp. 337–346.

Wiebe, J. and E. Riloff (2005). Creating subjective and objective sentence classifiers from unannotated texts. In *Proceedings of Sixth International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2005)*, Mexico City, Mexico.

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

Wilson, T., J. Wiebe, and P. Hoffmann (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the 2005 Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP-05)*, Vancouver, Canada.

Wilson, T., J. Wiebe, and R. Hwa (2006). Recognizing strong and weak opinion clauses. *Computational Intelligence 22*(2), 73–99.

Witten, I. H. and E. Frank (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (2nd ed.). San Francisco: Morgan Kaufmann.

Zwarts, F. (1995). Nonveridical contexts. *Linguistic Analysis 25*(3/4), 286–312.