Impact of Lexical Filtering on Overall Opinion Polarity Identification

Franco Salvetti[†] **Stephen Lewis**[‡] **Christoph Reichenbach**[†]

franco.salvetti@colorado.edu stephen.lewis@colorado.edu christoph.reichenbach@colorado.edu

† Department of Computer Science, University of Colorado at Boulder
430 UCB, Boulder, CO 80309-0430

‡ Department of Linguistics, University of Colorado at Boulder
295 UCB, Boulder, CO 80309-0295

Abstract

One approach to assessing overall opinion polarity (OvOP) of reviews, a concept defined in this paper, is the use of supervised machine learning mechanisms. In this paper, the impact of lexical filtering, applied to reviews, on the accuracy of two statistical classifiers (Naive Bayes and Markov Model) with respect to OvOP identification is observed. Two kinds of lexical filters, one based on hypernymy as provided by WordNet (Fellbaum 1998), and one hand-crafted filter based on part-of-speech (POS) tags, are evaluated. A ranking criterion based on a function of the probability of having positive or negative polarity is introduced and verified as being capable of achieving 100% accuracy with 10% recall. Movie reviews are used for training and evaluation of each statistical classifier, achieving 80% accuracy.

Introduction

The dramatic increase in use of the Internet as a means of communication has been accompanied by an increase in freely available online reviews of products and services. Although such reviews are a valuable resource to customers who want to make well-informed shopping decisions, their abundance and the fact that they are mixed in terms of positive and negative overall opinion polarity are often obstacles. For instance, a customer that is already interested in a certain product may want to read some negative reviews just to pinpoint possible drawbacks, but has no interest in spending time reading positive reviews. In contrast, customers interested in watching a good movie may want to read reviews that express a positive overall opinion polarity. The overall opinion polarity of a review, with values expressed as positive or negative, can be represented through the classification that the author of a review would assign to it, if requested. Such a classification is here defined as the overall opinion polarity (OvOP) of a review, or simply the polarity. The process of identifying OvOP of a review will be referred to as Overall Opinion Polarity Identification (OvOPI).

A system that is capable of labelling a review with its polarity is valuable for at least two reasons. First, it allows the reader interested exclusively in positive (or negative) reviews to save time by reducing the number of reviews to be read. Second, since it is not uncommon for a review that starts with positive polarity to turn out to be negative, or vice versa, it avoids the risk of a reader erroneously discarding a review just because it first appears to have the wrong polarity.

Copyright © 2004, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

In this paper we frame a solution to OvOPI based on a supervised machine learning approach. In such a framework we observe the effects of lexical filtering, applied to reviews, on the accuracy of two statistical classifiers trained on such filtered data. We have implemented two different kinds of lexical filters, one based on hypernymy as provided by WordNet (Fellbaum 1998), and one based on part-of-speech (POS) tags.

The results obtained by experiments based on movie reviews revealed that WordNet filters produce less improvement than do POS filters, and that for neither is there evidence of significantly improved performance over the system without filters, although the overall performance of our system is comparable to systems in current research, achieving an accuracy of 81%.

In the domain of OvOPI of reviews it is often acceptable to sacrifice recall for accuracy. Here we also present a system whereby the reviews are ranked based on a function of the probability of being positive/negative. Using this ranking method we achieve 100% accuracy when we accept a recall of 10%. This result is particularly interesting for applications that rely on web data, because the customer is not always interested in having all the possible reviews, but many times is interested in having just a few positive and a few negative. From this perspective accuracy is more important than recall.

Related Research

Research has demonstrated that there is a strong positive correlation between the presence of adjectives in a sentence and the presence of opinion (Wiebe, Bruce, & O'Hara 1999). Hatzivassiloglou et al., in (Hatzivassiloglou & McKeown 1997), combined a log-linear statistical model that examined the conjunctions between adjectives, (such as "and", "but", "or"), with a clustering algorithm that grouped the adjectives into two sets which were then labelled positive and negative. Their model predicted whether adjectives carried positive or negative polarity with 82% accuracy. However, because the model was unsupervised it required an immense, 21 million word corpus to function.

Turney extracted n-grams based on adjectives (Turney 2002). In order to determine if an adjective had a positive/negative polarity he used AltaVista and its function NEAR. He combined the number of co-occurrences of the adjective under investigation NEAR the adjective 'excellent' and NEAR the adjective 'poor' thinking that high occurrence NEAR 'poor' implies negative polarity and high occurrence NEAR 'excellent' implies positive polarity. Turney achieved an average of 74% accuracy in OvOPI across all domains. The performance on movie reviews, however, was especially poor at only 65.8%, indicating that OvOPI for movie reviews is a more difficult task than for other product reviews.

Pang et al. concluded that the task of polarity classification was not the same as topic classification (Pang, Lee, & Vaithyanathan 2002). They applied Naïve Bayes, Maximum Entropy and Support Vector Machine classification techniques to the identification of the polarity of movie reviews. They reported that the Naïve Bayes method returned a 77.3% accuracy using bigrams. Their best results came using unigrams, calculated by the Support Vector Machine at 82.9% accuracy. Maximum Entropy performed best using both unigrams and bigrams at 80.8% accuracy, and Naïve Bayes performed best at 81.5% using unigrams with POS tags.

Statistical approaches to polarity identification

There are many possible approaches to identifying the actual polarity of a document. Our analysis uses statistical methods, namely supervised machine learning, to identify the likelihood of reviews having "positive" or "negative" polarity with respect to previously hand-classified training data. These methods are fairly standard and well-understood; we list them below for the sake of completeness:

Naïve Bayes Classifier

In this paper the "features" used to develop Naïve Bayes are referred to as "attributes" to avoid confusion with text "features." In our approach, all word/POS-tag pairs that appear in the training data are collected and used as attributes. Our implementation of Naïve Bayes (see e.g. (Russell & Norvig 2003), p482). One interesting aspect of our particular application of Naïve Bayes is that we consider both the probabilities of attributes being present and the probabilities of them not being present. As most attributes do not appear in a test review, this means that most factors in the product probability are based on what is not written in it. This is one major difference from Markov Model classifiers.

Classifier based on Markov Models

Because the Naïve Bayes classifier defined in the previous section builds probabilistic models based on individual occurrences of words, it is provided with relatively little information regarding the phrasal structure. Markov Models ((Russell & Norvig 2003), p538), however, do capture this information. As such, we implemented a classifier based on two Markov language models: One for positive, and another for negative reviews.

Features for analysis

Statistical analysis depends on a sequence of tokens it uses as characteristical features of the objects it attempts to analyze; the only necessary property of these features is that it must be possible to identify whether two features are equal.

The most straightforward way of dealing with the information we find within reviews would be to use individual words from the review data as tokens. However, just using the words discards semantic information about the remainder of the sentence; as such, it may be desirable to first perform some sort of semantic analysis to enrich the tokens with useful information, or even discard misleading or irrelevant information (noise), in order to increase accuracy.

Three basic approaches for handling this kind of data preprocessing come to mind:

- Leave the data as-is: Each word will be represented by itself
- Parts-of-speech tagging: Each word is enriched by a POS tag, as determined by a standard tagging technique (such as the Brill Tagger (Brill 1995))
- Perform POS tagging and parse (using e.g. the Penn Treebank (Marcus, Santorini, & Marcinkiewicz 1994))

Unfortunately, the third approach not only had severe performance issues during our early experiments, but also raises conceptional questions of how such data would be incorporated into a statistical analysis. We thus focus our analysis in this paper on POS-tagged data (sentences consisting of words enriched with information about their parts of speech), which seems to be a good candidate for a worthwhile source of information, for the following reasons:

1. As discussed in (Losee 2001), information retrieval with POS-tagged data improves the quality of an analysis in many cases,
2. It is a computationally inexpensive way of increasing the amount of (potentially) relevant information,
3. It gives rise to POS-based filtering techniques for further refinement, as we discuss below.

We thus make the following assumptions about our test and training data:

1. All words are transformed into upper case,
2. All words are stemmed,
3. All words are transformed into (word, POS) tuples by POS tagging (notation word / POS).

All of these are computationally easy to achieve (with a reasonable amount of accuracy) using the Brill Tagger.

Experiments

Settings

- Data: taken from Cornell Data (Pang, Lee, & Vaithyanathan 2002)
- Part-of-speech tagger: Brill tagger (Brill 1995)
- WordNet: WordNet version 1.7.13 (Fellbaum 1998)

Movie reviews are used for training and evaluation of each statistical classifier. The decision to use only movie reviews for training and test data was based on the fact that OvOPI of movie reviews is particularly challenging as shown in (Turney 2002), and therefore can be considered a good environment for testing any system designed for OvOPI. The other reason for using movie reviews is the availability of large bodies of free data on the web. Specifically we used the data available through Cornell University from the Internet Movie Database. The Cornell data consists of 27,000 movie reviews in HTML form, using 35 different rating scales such as A...F or 1...10 in addition to the common 5 star system. We divided them into two classes (positive and negative) and took 100 reviews from each class as the test set. For training sets, we first identified the reviews most likely to be positive or negative. For instance, when reviews contained letter grade ratings, only the A and F reviews were selected. This was done in an attempt to minimize the effects of conflicting polarities. From these reviews, we randomly sampled from 50 to 750 (in increments of 50) reviews from the remaining reviews in each class. This resulted in training set sizes of 100, 200, ..., 1500 (in increments of 100). HTML documents were converted to plain text, tagged using the Brill tagger, and fed into filters and classifiers. The particular combinations of filters and classifiers and their results are described in the following sections.

The fact that as a training set we used data labelled by a reader and not directly by the writer poses a potential problem. We are learning a function that has to mimic the label identified by the writer, but we are using data labelled by the reader. We assume that this is an acceptable approximation because there is a strong practical relation between the label identified by the original writer and the reader. The authors themselves may not have made the polarity classifications, but we assume that language is an efficient form of communication. As such, variances between author and reader classification should be minimal.

Naïve Bayes

According to linguistic research, adjectives alone are good indicators of subjective expressions (Wiebe 2000). Therefore, determining semantic orientation by analyzing occurrences of individual adjectives in a text should be an effective method. To identify the semantic orientation of movie reviews, a Naïve Bayes classifier using adjectives is a promising model. The effectiveness of adjectives compared to other parts-of-speech is evaluated by applying and comparing the results on data with only adjectives against data with all parts-of-speech. The impact of at-level generalization from adjectives to synsets is also measured. The Naïve Bayes classifier described above was applied to:

1. tagged data
2. data containing only the adjectives
3. data containing only the synsets of the adjectives

The adjectives in 3 were generalized to at-level synsets (or "Sets of Synonyms"; see "WordNet filtering", below) using a combination of the POS filter module and the generalization filter module. For each training data set, add-one smoothing was applied to the Naïve Bayes classifier. Table 1 shows the resulting accuracies of each data set type and size. The results indicate that at-level generalization of adjectives is not effective and that extracting only adjectives degrades the classifier. However, this does not imply that filtering does not work. Adjectives constitute 7.5% of the text in the data. The accuracy achieved on such a small portion of the data indicates that a significant portion of the SO information is carried in the adjectives alone. Although the resulting accuracies are better in all-POS data, adjectives can still be considered good clues of semantic orientation.

Size	All-POS	JJ	JJ+WN
100	.615	.640	.650
200	.740	.670	.665
300	.745	.700	.690
400	.740	.700	.730
500	.740	.705	.705
600	.760	.710	.670
700	.775	.715	.710
800	.765	.715	.700
900	.785	.725	.710
1000	.765	.755	.720
1100	.785	.750	.760
1200	.765	.734	.750
1300	.775	.730	.710
1400	.775	.735	.745
1500	.795	.730	.735

Table 1: Accuracies of Naïve Bayes classifier. JJ means "adjectives only", WN indicates synset mapping using the generalization filter.

Markov Model

Three types of data are applied to the Markov Model classifiers described previously:

1. Tagged data without any filtering,
2. Tagged data with POS filters,
3. Tagged data with both POS filters and generalization filters.

Witten-Bell smoothing is applied to this classifier.

Part of Speech Filters

Careful analysis of movie reviews has made it clear that even the most positive reviews have portions with negative polarity or no clear polarity at all. Since the training data used here consists of complete classified reviews, the presence of parts with conflicting polarities or lack of polarity within a review presents a major obstacle for accurate OvOPI. As illustration of this inconsistent polarity, the following were all taken from a single review[1].

[1]APOLLO 13, A film review by Mark R. Leeper, Copyright ©1995 Mark R. Leeper

(1a)	Copula Conversion	`is/*` → `*/COP`
(1b)	Negation conversion	`not/*` → `/NEG`
(2)	Noun generalization	`*/NN` → `/NN`
(3)	POS Tossing	`*/CC` → ∅

Figure 1: Abbreviated filter rule specification (illustrative details only)

"Special effects are first-rate"
(positive polarity)
"The character is written thinly"
(negative polarity)
"The scenes were shot in short segments"
(no clear polarity)

This observation can be taken to lower levels as well. Individual phrases and words vary in their contribution to opinion polarity. It may even be said that only some part of the meaning of a word contributes to opinion polarity (see WordNet filter section). Any portion that does not contribute to the OvOP is noise. To reduce noise, filters were developed that use POS tags to do the following.

1. Introduce custom parts of speech when the tagger does not provide desired specificity (negation and copula).
2. Remove the words that are least likely to contribute to the polarity of a review (determiner, preposition, etc.)
3. Reduce parts of speech that introduce unnecessary variance to POS only. It may be useful, for instance, for the classifier to record the presence of a proper noun. However, to include individual proper nouns would unnecessarily decrease the probability of finding the same n-grams in the test data.

Experimentation involved multiple combinations of such filter rules, yielding several separate filters. An example of a specification of POS filter rules is shown in Figure 1.

The POS filters are not designed to reduce the effects of conflicting polarity. They are only designed to reduce the effect of lack of polarity. The effects of conflicting polarity have instead been addressed by careful preparation of the training data, as mentioned earlier.

One design principle of the filter rules is that they filter out parts of speech that do not contribute to the semantic orientation and keep the parts of speech that do contribute such meaning. Based on analysis of movie review texts, we devised "filter rules" that take Brill-tagged text as input and return less noisy, more concentrated sentences that have a combination of words and word/POS-tag pairs removed from the original. A summary of the filter rules defined in this experiment is shown in Table 2.

Wiebe et al., as well as other researchers, showed that subjectivity is especially concentrated in adjectives (Wiebe, Bruce, & O'Hara 1999; Department ; Turney & Littman 2003). Therefore, no adjectives or their tags were removed, nor were copula verbs or negative markers. However, noisy information such as determiners, foreign words, prepositions, modal verbs, possessives, particles, interjections, etc. were removed from the text stream. Other parts of speech,

POS[1]	r_1	r_2	r_3	r_4	r_5
JJ[2]	K	K	K	K	K
RB[3]	D	K	K	K	K[4]
VBG	K	K	K	K	D
VBN	K	K	K	K	D
NN[5]	G	G	G	G	G
VBZ	D	D	K	K	D
CC	D	D	D	K	K
COP[6]	K	K	K	K	K

K: Keep D: Drop G: Generalize

[1]Abbreviations of POSs are based on the tree bank's notation
[2]JJ includes JJ, JJR and JJS
[3]RB includes RB, RBR and RBS except "not"
[4]RBRs are dropped and RB and RBS stay
[5]NN and NNS are generalized to NN, and NNP and NNPS are generalized to NNP
[6]COPS are particular verbs: is, was, am, are, were, be, been, like, liked, dislike, disliked, hate, hated, seem and seemed

Table 2: Summary of POS filter rules

such as nouns and verbs, were removed but their POS-tags were retained. The output returned from the filter did not keep the original sentence structure. The concrete POS filtering rules applied in this experiment are shown in Table 2. The following is an example of the sentence preprocessing:

- All Steve Martin fans should be impressed with this wonderful new comedy
- /NNP /NNP /NN be/COP /VBN wonderful/JJ new/JJ /NN

The resulting accuracies on POS filter rules and different sizes of data sets are listed in Table 3.

WordNet filtering

In non-technical written text it is uncommon to encounter repetitions of identical words; this is generally considered "bad style". As such, many authors attempt to use synonyms for words whose meanings they need often, propositions, or even generalizations. We attempted to address two of these perceived issues by identifying words with a set of likely synonyms, and by *hypernymy generalization*. For the implementation of these techniques, we took advantage of the WordNet (Fellbaum 1998) system, which provides the former by means of *synsets* for four separate classes of words (verbs, nouns, adjectives and adverbs), and the latter through *hypernymy relations* between synsets of the same class.

Synonyms

WordNet maps each of the words it supports into a synset, which is an abstract entity encompassing all words with a "reasonably" similar meaning. In the case of ambiguous words, multiple synsets may exist for a word; in these instances, we picked the first one.

Note that synonyms (and general WordNet processing) are only available in instances where the word under consideration falls in one of the four classes of words we outlined

size	r_1	r_2	r_3	r_4	r_5	All-POS
100	.555	.625	.625	.630	.630	.575
200	.675	.710	.710	.700	.700	.655
300	.660	.635	.635	.655	.655	.675
400	.700	.660	.660	.685	.685	.710
500	.640	.665	.665	.680	.680	.720
600	.685	.750	.750	.765	.765	.745
700	.705	.700	.700	.690	.690	.735
800	.700	.740	.740	.715	.715	.690
900	.700	.740	.740	.765	.765	.760
1000	.730	.745	.745	.730	.730	.765
1100	.750	.745	.745	.715	.715	.775
1200	.710	.710	.710	.720	.720	.765
1300	.715	.695	.695	.705	.705	.770
1400	.755	.745	.745	.755	.755	.805
1500	.725	.730	.730	.750	.750	.770

Table 3: Accuracies on POS filtering for the various rules.

above. We determined the appropriate category for each word by examining the tag it was assigned by the Brill tagger, not touching words which fell outside of these classes.

Hypernyms

For verbs and nouns, WordNet provides a hypernymy relation, which can be informally described as follows: Let s_1, s_2 be synsets. Then s_1 is hypernym of s_2, notation $s_1 \succ s_2$, if and only if anything that can be described by a word in s_2 can also be described by a word in s_1, and $s_1 \neq s_2$.

For each of the hypernym categories, we determine a set of abstract synsets $\mathcal{A}$ such that, for any $a \in \mathcal{A}$, there does not exist any s such that $s \succ a$.

We say that a synset h is a *Level n hypernym* of a synset s if and only if $h \succ^* s$ and one of the following holds for some $a \in \mathcal{A}$:

1. $a \succ^n h$
2. $s = h$ and $a \succ^l s$, with $l < n$

For example, given the WordNet database, a hypernym generalization of level 4 for the nouns "movie" and "performance" will generalize both of them to one common synset which can be characterized by the word "communication."

Analysis

In order to determine the effects of translating words to synsets and performing hypernymization on them, we ran a series of tests on them, which quickly determined that the effects of pure synset translation were negligible. We thus experimented with the computation of level n hypernyms with $n \in \{0 \ldots 10\}$, separately for nouns and verbs.

Our measurements showed that applying hypernym generalization to classifiers trained on large data sets caused a degradation in the quality of our classification due to a loss of information. Apparently, bigram classification is already capable of making use of the more fine-grained information gathered from reasonably-sized (1500+1500 reviews) training sets. For very small data sets (50 reviews and less), however, we observed an absolute improvement of 2.5% in comparison both to full generalization and no generalization at all. Increasing the size of the set of observable events by using trigram models resulted in a small gain (around 1%). Interestingly, the effect of verb generalization was relatively small in comparison to noun generalization for similar hypernymy levels.

We assume that the lack of improvement for large training corpora is due to at least the following reasons:

- WordNet is *too general* for our purposes: It considers many meanings and hypernymy relations which are rarely relevant to the field of Movie Reviews, but which potentially take precedence over other relations which might be more appropriate here.
- Choosing the first synset out of the set of choices is unlikely to yield the correct result, given the lack of WordNet's specialization on our domain of focus.
- For reasonably large data sets, supervised learning mechanisms gain sufficient confidence with related words to make this particular auxiliary technique less useful.

Considering this, the use of a domain-specific database seems to be a promising approach to improving our performance for this technique.

Selection by Ranking

The probabilistic models computed by the Naïve Bayes classifiers were sorted by log posterior odds on positive and negative orientations for the purpose of ranking, i.e. by a "score" computed as follows:

$$\text{score} = \log \Pr(+|\texttt{rv}) - \log \Pr(-|\texttt{rv})$$

where

- $\texttt{rv}$ is the review under consideration,
- $\Pr(+|\texttt{rv})$ is the probability of $\texttt{rv}$ being a review of positive polarity,
- $\Pr(-|\texttt{rv})$ analogously is the probability of the review being of negative polarity.

We modified the classifier so that it:

1. Sorts the reviews by log posterior odds
2. Returns the first N reviews as positive results
3. Returns the last N reviews as negative results

The resulting accuracies and recalls on different N are summarized in Table 4. The classifier was trained on the same 1500 review data set and was used with ranking on a repository of 200 reviews which were identical to the test data set. The result is very positive and indicates that adjectives provide enough sentiment to detect extremely positive or negative reviews with good accuracy. While the number of reviews returned is specified in this particular example, it is also possible to use assurance as the cutoff criterion by using posterior odds.

N	precision	recall
10	1.000	.100
20	.975	.195
30	.900	.270
40	.900	.360
50	.880	.440
60	.867	.520
70	.830	.580
80	.780	.625
90	.780	.680

Table 4: Precisions and Recalls by Number of Inputs

Discussion

Taking all results into consideration, both the Naïve Bayes classifier and Bigram Markov Model classifier performed best when trained on sufficiently large data sets without filtering. For both Bigram and Trigram Markov Models, we observed a noticeable improvement with our generalization filter when training on very small data sets; for trigram models, this improvement even extended to fairly large data sets (1500 reviews).

One explanation for this result is that the filters are unable to make use of the more fine-grained information provided to them. A likely reason for this is that the ratio between the size of the set of observable events and the size of the training data set is comparatively large in both cases. However, further research and testing will be required in order to establish a more concrete understanding of the usefulness of this technique. The learning curve of classifiers with the POS filter and/or the generalization filter climbs at higher rates than those without the filters and results in lower accuracy with larger data sets. One possible explanation for the higher climbing rates is that the POS filter and the generalization filter compact the possible events in language models while respecting the underlying model by reducing vocabulary. This also explains why the plateau effect is observed with smaller data set sizes. The degraded results with filters also indicate that by removing information from training and test data, the compacted language model loses resolution.

Conclusion

A framework of two-phased classification mechanism is introduced and implemented with a POS filter, a generalization filter, a Naïve Bayes classifier and a Markov Model classifier. Accuracies of combinations of filters and classifiers are evaluated by experiments. Although the results from classifications without filters are better than those with filters, the POS filters and generalization filters are observed to still have potential to improve overall opinion polarity identification. Generalization filtering using WordNet shows good accuracy for small data sets and warrants further research. Using the Naïve Bayes classifier with ranking on adjectives confirmed that desired precision can be achieved by sacrificing recall. For the task of finding reviews of strong positive or negative polarity within a given data set, very high precision was observed for adequate recall.

Acknowledgements

The authors would like to thank Tomohiro Oda for his extensive help and support. Further acknowledgements go to Larry D. Blair, Assad Jaharria, Helen Johnson, Jim Martin, Jeff Rueppel and Philipp Wetzler for their valuable contributions.

References

Brill, E. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics* 21(4):543–565.

Department, V. H. Effects of adjective orientation and gradability on sentence subjectivity.

Fellbaum, C. 1998. Wordnet: An electronic lexical database.

Hatzivassiloglou, V., and McKeown, K. R. 1997. Predicting the semantic orientation of adjectives. In Cohen, P. R., and Wahlster, W., eds., *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, 174–181. Somerset, New Jersey: Association for Computational Linguistics.

Losee, R. M. 2001. Natural language processing in support of decision-making: phrases and part-of-speech tagging. *Information Processing and Management* 37(6):769–787.

Marcus, M. P.; Santorini, B.; and Marcinkiewicz, M. A. 1994. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics* 19(2):313–330.

Pang, B.; Lee, L.; and Vaithyanathan, S. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Russell, S., and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach (second edition)*. Prentice-Hall, Englewood Cliffs, NJ.

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

Turney, P. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, 417–424.

Wiebe, J.; Bruce, R. F.; and O'Hara, T. 1999. Development and use of a gold-standard data set for subjectivity classifications. In *Proc. 37th Annual Meeting of the Assoc. for Computational Linguistics (ACL-99)*.

Wiebe, J. 2000. Learning subjective adjectives from corpora. In *AAAI/IAAI*.