Sentiment Classification of Movie and Product Reviews Using Contextual Valence Shifters

Alistair Kennedy and **Diana Inkpen**
University of Ottawa, Ottawa, ON , K1N 6N5, Canada
{akennedy,diana}@site.uottawa.ca

Abstract

We present a method for determining the sentiment expressed by a customer review. The semantic orientation of a review can be positive, negative, or neutral. Our method counts positive and negative terms, but also takes into account contextual valence shifters, such as negations and intensifiers. Tests are done taking both negations and intensifiers into account, and also using only negations without intensifiers. Negations are used to reverse the semantic polarity of a particular term, while intensifiers are used to change the degree to which a term is positive or negative. We use the *General Inquirer* in order to identify positive and negative terms, as well as negations, overstatements, and understatements. We also test the impact of adding extra positive and negative terms from other sources, including a dictionary of synonym differences and a very large web corpus. To compute the corpus-based values of the semantic orientation of individual terms we use their association scores with a small group of positive and negative terms. We show that including contextual valence shifters improves the accuracy of the classification.

1 Introduction

In recent years a great deal of research has been done on categorizing documents. The categories could be based on subject, genre, or the sentiment expressed in the document. Sentiment classification (into positive or negative opinions) has many useful applications. One example is question answering. In cases where a user is asking an opinion question such as *What are the reasons for the US-Iraq war?* will require the system to determine the perspective of the different sources, using sentiment classification (Yu and Hatzivassiloglou, 2003). Another application of sentiment classification is text summarization. If a program can pick out the sentiment of a review, it can used it to label the review; this could be an important part of the process of summarizing reviews (Pang et al., 2002).

Two approaches to classifying sentiment are compared in this paper. The first approach is to count positive and negative terms in a review, where the review is considered positive if it contains more positive than negative terms, and negative if the number of negative terms is greater than the number of positive terms. A review is neutral if it contains an equal number of positive or negative terms.

It should be noted that our method of term counting is not as effective as a Machine Learning algorithm. In (Pang et al., 2002) it is shown that using a Machine Learning algorithm outperforms a simple term counting method. Our goal is not to show that a term counting method can perform as well as a Machine Learning method, but to measure the impact of valence shifters on sentiment classification. The term counting method can be easily modified to use valence shifters. With a machine learning algorithm it could be difficult to incorporate valence shifters in a way that makes it clear if the improvement in the results is caused by the use of valence shifters or by some other factors.

Positive and negative terms are initially taken from the *General Inquirer* (Stone et al., 1966) (hereafter GI), which is a dictionary that contains information about English word senses, including tags that label them as positive, negative, negations, overstatements or understatements.

The second method counts positive and negative terms, but takes contextual valence shifters into account. Valence shifters are terms that can change the semantic orientation of another term. Basically this means we are looking for terms that switch a positive term to negative and vice versa. This includes such terms as *not*, *never*, *none*, *nobody*, etc. (Polanyi and Zaenen, 2004). Terms that change the intensity of the positive or negative term are also examined. These terms increase of decrease the weight of a positive or negative term.

Our basic system uses the first method, while our improved system uses the second method. For both systems we also add more positive and negative terms from several other sources.

2 Background and Related Work

Sentiment classification of reviews has been the focus of recent research. It has been attempted in different domains such as movie reviews, product reviews, and customer feedback reviews (Gamon, 2004; Pang et al., 2002; Pang et al., 2004; Turney and Littman, 2003). Much of the research up to this point has focused on training machine learning algorithms such as Support Vector Machines (SVMs) to classify reviews. Research has also been done on positive/negative term counting methods and automatically determining if a term is positive or negative (Turney and Littman, 2002).

2.1 Machine Learning for Determining Sentiment

One of the most common methods of classifying documents into positive and negative terms is to train a Machine Learning algorithm to classify the documents. Several ML algorithms are compared in (Pang et al., 2002; Pang et al., 2004) where it was found that SVMs generally gave better results. Unigrams, bigrams, part of speech information, and the position of the terms in the text were used as features; however only using unigrams were found to give the best results. This method was found to be up to 83% accurate.

Bayesian belief networks have also been used to determine the sentiment of a document. Bayesian belief networks were used to represent a Markov Blanket (Bai et al., 2004), which is a directed acyclic graph where each vertex represents a word and the edges are dependencies between the words. The Bayesian belief network is then reordered using Tabu Search.

Sentiment classification has also been done on customer feedback reviews (Gamon, 2004). A variety of features are used on SVMs in an attempt to divide the data set not only into positive and negative, but to give rankings of 1, 2, 3 and 4 where 1 means "not satisfied" and 4 means "very satisfied". The proposed system was fairly good at distinguishing classes 1 from 4, with about 78% accuracy. Separating classes 1,2 from 3,4 was more difficult and was only

69% accurate. These results were achieved when using the top 2000 features selected by log likelihood ratios.

2.2 Distinguishing Objective from Subjective Statements

Methods for extracting subjective expressions from corpora are presented in (Wiebe et al., 2004). Subjectivity clues include low-frequency words, collocations, and adjectives and verbs identified using distributional similarity. In (Riloff and Wiebe, 2003) a bootstrapping process learns linguistically rich extraction patterns for subjective expressions. High-precision classifiers label unannotated data to automatically create a large training set, which is then given to an extraction pattern learning algorithm. The learned patterns are then used to identify more subjective sentences.

A method of distinguishing objective statements from subjective statements in presented in (Pang et al., 2004). This method is based on the assumption that objective and subjective sentences are more likely to appear in groups. First, each sentence is given a score indicating if the sentence is more likely to be subjective or objective using a Naïve Bayes classifier trained on a subjectivity data set. The system then adjusts the subjectivity of a sentence based on how close it is to other subjective/objective sentences. This method was found to produce results with up to 86% accuracy on the movie review data set.

A similar experiment is presented in (Yu and Hatzivassiloglou, 2003). A Naïve Bayes classifier is used to discover opinion sentences by training it on a labeled data set. They also combine multiple Naïve Bayes classifiers together for the same task, where each Naïve Bayes classifier focuses on a different part of the feature set. The feature sets included unigrams, bigrams, trigrams, part of speech information, and polarity. Once it was discovered if a sentence is objective or subjective, a list of positive and negative terms was used to determine the sentiment of the sentence. This experiment was done as a starting point towards answering opinion questions.

2.3 Negative and Positive Short Stories

Not only reviews can be classified as positive or negative. It is possible to classify the tone of short stories as being either positive or negative. In (Bolasco and della Ratta-Rinaldi, 2004) a system for determining when a short story has a negative or positive tone is presented. To do this, a data set of 2000 short stories written by Italian students of both sexes and varying age groups was collected. Terms were then translated into English and the *General Inquirer* was used to determine if the translated terms are positive or negative. On average the stories were found to contain more negative terms than positive terms (Bolasco and della Ratta-Rinaldi, 2004).

2.4 Determining Sentiment

Research on prediction the semantic orientation of adjectives was initiated by (Hatzivassiloglou and McKeown, 1997). An unsupervised learning algorithm is used in (Turney, 2002; Turney and Littman, 2003) to determine the semantic orientation of individual terms. The algorithm starts with 7 known positive terms and 7 known negative terms. The algorithm takes a search term and uses AltaVista's NEAR operator to find how many documents have the search term near the 7 positive terms and the 7 negative terms. The difference in Pointwise Mutual Information (PMI) score with the two sets is then used to determine the SO-PMI score, which gives the degree to which each term is positive or negative (Turney and Littman, 2002). The PMI score of two words w_1 and w_2 is given by the probability of the two words occurring together divided

by the probabilities of each work in part:

$$\text{PMI}(w_1, w_2) = log\frac{P(w1, w2)}{P(w_1)P(w_2)} = log\frac{hits(w_1, w_2)N}{hits(w_1)hits(w_2)}$$

The formula for the semantic orientation of a word can be expressed as:

$$\text{SO-PMI}(word) = PMI(word, p\_query) - PMI(word, n\_query)$$

where the positive and negative reference terms are:

$p\_query = good$ OR $nice$ OR $excellent$ OR $positive$ OR $fortunate$ OR $correct$ OR $superior$
$n\_query = bad$ OR $nasty$ OR $poor$ OR $negative$ OR $unfortunate$ OR $wrong$ OR $inferior$

OR and NEAR are operators offered by the AltaVista search engine (NEAR is no longer supported). By approximating the PMI values using number of hits returned by the search engine and ignoring the number of documents in the corpus (N), the formula becomes:

$$\text{SO-PMI}(word) = \log\frac{hits(word \text{ NEAR } p\_query)\ hits(n\_query)}{hits(word \text{ NEAR } n\_query)\ hits(p\_query)}$$

The semantic orientation of bigrams can also be determined (Turney, 2002). The semantic orientation of terms and phrases can be used to determine the sentiment of complete sentences and reviews. 410 reviews from epinions.com were taken and the accuracy of classifying the documents was found when computing the sentiment of phrases for different kinds of reviews. Results ranged from 84% for automobile reviews to as low as 66% for movie reviews (Turney, 2002).

3 The Data Sets

We use two data sets to test our method of reviews classification. The first data set is a set of 140 reviews taken from epinions.com. This data set contains 70 positive and 70 negative reviews. The reviews were collected from a variety of different products, including air conditioners, sewing machines, vacuum cleaners, TVs, cookware, beer and wine. Reviews at epinions.com are rated with a 5 star system where 1 is low and 5 is high. Reviews where the product gets 1 or 2 stars are assumed to be negative, reviews with 4 or 5 stars are assumed to be positive. The second data set is the latest version of the movie review data set used in (Pang et al., 2002; Pang et al., 2004). This data set contains 2000 movie reviews, 1000 positive and 1000 negative.

4 Methodology

This section outlines the steps taken in creating a system to classify sentiment in reviews. The basic idea behind this system is to classify reviews based on how many positive and negative terms are present in a document. If there are more positive than negative terms then it is considered to be positive. If there are more negative than positive terms it is considered to be negative. If there are equal numbers of positive and negative terms it is neutral. This idea of counting positive and negative terms and expressions was proposed by (Turney, 2002). We augment this method by taking contextual valence shifters into account.

FUN#1	H4Lvd Positiv Pstv Pleasur Exprsv WlbPsyc WlbTot Noun PFREQ 97% noun-adj: Enjoyment, enjoyable
FUN#2	H4Lvd Negativ Ngtv Hostile ComForm SV RspLoss RspTot SUPV 3% idiom-verb: Make fun (of) – to tease, parody

Figure 1: GI entries for the word *fun*.

NOT	H4Lvd Negate NotLw LY adv: Expresses negation
FANTASTIC	H4Lvd Positiv Pstv Virtue Ovrst EVAL PosAff Modif
BARELY	H4Lvd Undrst Quan If LY

Figure 2: GI entries for the words *not*, *fantastic* and *barely*.

Identifying Positive and Negative Terms The main resource used for identifying positive and negative terms is the *General Inquirer*[1] (Stone et al., 1966). GI is a system which lists terms as well as different senses for the terms. For each sense it provides a short definition as well as other information about the term. This includes tags that label the term as being positive, negative, a negation term, an overstatements, or an understatements. For example, there are two senses of the word *fun* as seen in Figure 1. One sense is a noun or adjective for *enjoyment* or *enjoyable*. The second sense is a verb that means *to ridicule or tease*, *to make fun of*. The first sense of the word is positive, marked as *Positiv*, while the second is negative, marked as *Negativ*. There are other labels for each sense. It is also indicated that the first sense occurs 97% of the times while the second sense occurs only 3% of the times.

We also examine negations, overstatements and understatements. Figure 2 shows examples of the words *not*, *fantastic* and *barely* which are examples of *negation*, *overstatement* and an *understatement*. In the case of these three terms there was only one sense each.

The *General Inquirer* has 1915 positive senses and 2291 negative senses. We add extra positive and negative senses from *Choose the Right Word* (Hayakawa, 1994) (hereafter CTRW), obtaining 1955 positive senses and 2398 negative senses. There are 696 overstatements and 319 understatements in GI. CTRW is a dictionary of synonyms, which lists nuances of lexical meaning, extracted in (Inkpen et al., 2004). When we add overstatements and understatements from CTRW, there are 1269 overstatements and 412 understatements.

Positive and negative terms were also found in other sources. An example of a negative statement from CTRW is *smugness*, while a positive example is *soothing*. Both of these terms are not found in GI. A list of adjectives with positive and negative senses is also used as a source of words (Taboada and Grieve, 2004). From this list of adjectives we find negative terms such as *whiney* and positive terms such as *trendy*. When using SO-PMI scores to discover new positive and negative terms there are no senses of terms. Names such as *Hitler* and *Saddam* are found to be negative using SO-PMI. An example of a positive word found using SO-PMI is *happily* which does not appear in GI, CRTW, or the list of adjectives.

Stemming One problem with this method of counting positive and negative terms is that we may need to remove the suffix of a given term in order to see if it exists in our list of terms. To do this we first examined the Porter stemming algorithm[2] (Porter, 1980). This algorithm does

[1] http://www.wjh.harvard.edu/~inquirer/homecat.htm
[2] http://www.tartarus.org/~martin/PorterStemmer/

not produce a lemma for a term, but rather maps similar words to a string. For example *wrongly* becomes *wrongli* when used with Porter's algorithm. To get around this issue we execute the following steps when before using Porter's stemming algorithm: (1) check to see if the term is in our list of terms; (2) replace all words ending in *ily*, *ies*, and *ied* with *y*; (3) remove all *ly*, *ing*, *d*, *s*, *r*, and *ity* from the end of the original word; (4) use Porter's algorithm to stem the original word. At every step it checks to see if the term exists in its new form in our list of terms. If the term is found, it does not perform the next step.

Word Sense Disambiguation There are many terms that have multiple meanings. These terms have multiple definitions in GI. If a term is found for which there are many different definitions we may need to find out which definition corresponds to the correct sense. Often if one sense of a term is positive/negative, the other senses of the term will also be positive/negative. Instead of finding the sense in which a term is used we simply take all the senses and sum together the number of senses that are positive and negative. If there are more positive senses than negative we consider the term positive, if there are more negative senses than positive, we consider it negative, and if there is an equal number, or no positive/negative senses, then it is considered neutral. In GI there are only 15 words for which there are both positive and negative senses and only 12 words for which there are both overstatement and understatement senses. When adding terms from CTRW there are 19 words that have both positive and negative senses and 37 words that have both overstatement and understatement senses. Since theses numbers are fairly low in comparison with the total number of terms this should not have a significant impact on the results.

4.1 Incorporating Valence Shifters into the System

There are two different aspects of valence shifting that are used to improve our system. First, we take into account negations that can switch the sentiment of positive or negative terms in a sentence. Second, we take intensifiers into account. Intensifiers are terms that can change the degree to which a word is positive or negative (Polanyi and Zaenen, 2004). These valence shifters are incorporated into the system and tested to see if they give better results.

Negations Negations are terms that reverse the sentiment of a certain word (Polanyi and Zaenen, 2004). For example consider the sentence *This movie is good* versus *This movie is* ***not*** *good*. In the first one *good* is a positive term and so this sentence is positive. When *not* is applied to the clause, *good* is being used in a negative context and so the sentence is negative (Polanyi and Zaenen, 2004).

Intensifiers Intensifiers are terms that change the degree of the expressed sentiment. For example, in the sentence *This movie is* ***very*** *good*, the terms *very good* are more positive together than just *good* alone. Another example of an intensifier is *deeply* from the phrase ***deeply*** *suspicious*, which increases the intensity of the word *suspicious* (Polanyi and Zaenen, 2004). On another side, in the sentence *This movie is* ***barely*** *any good*, the term *barely* makes this statement less positive. Another term which decreases the intensity of a phrase is *rather* from the phrase ***rather*** *efficient* (Polanyi and Zaenen, 2004). These are examples of overstatements and understatements. Overstatements increase the intensity of a positive/negative term, while understatements decrease the intensity of that term. We note that the word *understatement* has other uses in linguistics (it could mean an entire clause or phrase.). Here we use it to mean a *diminisher* term. Terms that overstate and understate are also listed in GI. To allow for overstatements and understatements all positive sentiment terms in our system are given a value of

2. If they are preceded by an overstatement in the same clause then they are given a value of 3. If they are preceded by an understatement in the same clause then they are given a value of 1. Negative sentiment terms are given a value of -2 by default and -1 and -3 if preceded by understatements and overstatements respectively.

CTRW also contains a large number of terms, which are listed as having high strength or low strength. These strengths do not strictly mean that they are overstatements or understatements, however many of them can be used as such. We compared results when adding extra intensifiers from CTRW with only using the intensifiers from GI.

4.2 Experimental Setup and Results

We carried out experiments on both data sets, for both the basic and the improved system. The basic system simply counts positive and negative terms, while the improved system adds the treatment of contextual valence shifters. Also, several different dictionaries and word lists were used: the *General Inquirer* (GI); extra positive and negative terms, and extra overstatements and understatements from CTRW; list of positive/negative adjectives (Adj); and longer list of positive/negative terms (SO-PMI).

We used the accuracy of the classification, as well as precision, recall, and F-measure for each class in order to determine which system works best on each data set. The precision, recall, and F-measure show whether the loss in performance is for the positive or for the negative class.

First, we present in Table 1 results for the basic and improved systems when using only the terms in GI. Next, we present the results of the basic and improved systems when extra positive and negative terms and extra overstatements and understatements are added from CTRW.

We also added extra positive and negative adjectives from (Taboada and Grieve, 2004). The 1718 adjectives in this list come with semantic orientation scores based on hit counts collected through the AltaVista search engine. The scores are computed using Turney's method (Turney and Littman, 2002) (explained in section 2.4). We manually determined two thresholds: terms with SO-PMI value below 1.1 were labeled as negative, terms rated above 1.7 were labeled as positive. The terms with scores in between the two thresholds are considered neutral. In Table 1 this list of terms is denoted Adj.

In the last two versions of basic and improved system that we present in Table 1, we used a much longer list of positive and negative terms. We computed SO-PMI scores for all the 40000 content words in our datasets. To determine the SO-PMI scores we also used Turney's method, but instead of using AltaVista's NEAR operator (which is no longer available) we used the Waterloo MultiText System with a corpus of about one terabyte of text gathered by a Web crawler (Clarke and Terra, 2003). We collected co-occurrence counts in a window of 20 words. The formula is similar with the one from Section 2.4:

$$\text{SO-PMI}(word) = \log \frac{hits([20] > word \; .. \; p\_query) \; hits(n\_query)}{hits([20] > word \; .. \; n\_query) \; hits(p\_query)}$$

except that the NEAR operator is replaced with counts in a window of 20 words.

After we computed the SO-PMI scores, we used the positive/negative terms from GI to determine the best thresholds for the positive and negative terms. Terms with scores greater than 0.818 are positive, while terms with values less than -0.1845 are negative. This method gave a list of 4357 positive terms and 12633 negative terms, referred to as SO-PMI 1 in Table 1. We also tested this method with thresholds of 0.818 and -0.818, obtaining 4357 positive and 4291 negative terms – a more balanced ratio. This list is referred to as SO-PMI 2 in Table 1.

System	Review	Class	Accuracy	Precision	Recall	F-score
Basic: GI	Product	Positive	.679	.637	.929	.756
		Negative		.857	.429	.571
	Movie	Positive	.595	.578	.828	.681
		Negative		.698	.361	.475
Basic: GI & CTRW	Product	Positive	.679	.644	.929	.761
		Negative		.857	.429	.572
	Movie	Positive	.599	.581	.828	.683
		Negative		.702	.370	.485
Basic: GI & CTRW & Adj	Product	Positive	.667	.662	.754	.705
		Negative		.759	.586	.661
	Movie	Positive	.650	.645	.733	.686
		Negative		.696	.566	.624
Basic: GI & SO-PMI 1	Product	Positive	.671	.663	.843	.742
		Negative		.814	.500	.619
	Movie	Positive	.577	.879	.188	.310
		Negative		.546	.966	.698
Basic: GI & SO-PMI 2	Product	Positive	.600	.576	.971	.723
		Negative		.941	.228	.368
	Movie	Positive	.632	.611	.825	.702
		Negative		.735	.438	.549
Improved: GI	Product	Positive	.686	.637	.929	.755
		Negative		.912	.443	.596
	Movie	Positive	.627	.598	.817	.691
		Negative		.711	.436	.541
Improved: GI & CTRW	Product	Positive	.693	.641	.924	.757
		Negative		.886	.443	.591
	Movie	Positive	.627	.599	.824	.689
		Negative		.711	.443	.545
Improved: GI & CTRW & Adj	Product	Positive	.686	.671	.729	.699
		Negative		.703	.643	.672
	Movie	Positive	.667	.658	.734	.694
		Negative		.700	.601	.647
Improved: GI & SO-PMI 1	Product	Positive	.693	.663	.843	.742
		Negative		.872	.486	.624
	Movie	Positive	.584	.873	.200	.325
		Negative		.551	.968	.702
Improved: GI & SO-PMI 2	Product	Positive	.621	.588	.957	.738
		Negative		.870	.286	.430
	Movie	Positive	.651	.619	.816	.704
		Negative		.739	.486	.586

Table 1: Results for all systems. The basic system counts positive and negative terms. The improved system adds contextual valence shifters. Various lists of terms are used.

We note that the positive/negative labels computed with SO-PMI are not always reliable. For example, when looking at the SO-PMI scores of the words from our large list that are also in GI, the accuracy of the labeling them is 65% for the best possible thresholds (0.818 and -0.1845).

We also run tests that take negations into account, but not intensifiers. For movie reviews this method performed better than the basic system but worse than the improved system, while for product reviews the difference was very small. The results of this system are not included in this paper. For example, we did not show in Table 1 the results without the extra overstatements and understatements from CTRW, because these results were nearly identical to the one shown for the versions GI & CTRW. Therefore, we can say that negation terms contribute a lot, while overstatements and understatements have a lower impact.

Negations are applied to the first positive/negative term found after the negation term. If it is followed by a punctuation mark such as a quote or a period, the negation is not applied to anything. An alternative to this method is to apply the negation to all terms until then end of the sentence. We tried this method too, but it did not improve the results, it made them slightly worse. In future work we could parse the texts to get the exact scope of the negations, but we believe that the results would change the results very little. This is because negations that extend to two or three words are rare, and existing parsers may not detect them correctly.

5 Discussion of the Results

Our main goal was to determine how effective the addition of contextual valence shifters is to the simple method of counting positive and negative terms. From our experiments it is clear that the addition of valence shifters has an improving effect on the classification of reviews. It can be seen in Table 1 that the accuracy for both data sets with all dictionaries and word lists improves when contextual valence shifters were added. In most cases the F-measure also improves when contextual valence shifters are included.

To measure only the impact of the valence shifters we need to compare the basic system and the improved system when using the same list of positive/negative terms and overstatements/understatements. The improvement is statistically significant[3] for movie reviews, in all cases, while for the product reviews is not. For example, the gain of 3.2 percentage points (from 59.5% to 62.7%) between Basic: GI and Improved: GI from Table 1 is statistically significant at the level $\alpha = 0.05$.

Two other things that we examined are the effects of adding extra positive and negative terms, as well as the effects of adding extra overstatements and understatements. Adding extra positive and negative terms from CTRW generally improved the accuracy of the classification, for both data sets. This is true for both the basic and the improved system (with and without contextual valence shifters). For example Product reviews improve from 68.6% for Improved: GI to 69.3% for Improved GI & CTRW, while for movie reviews the accuracy remains constant at 62.7% for both the Basic and Improved systems. Adding extra overstatements and understatements from CTRW did not make much difference.

When we added a large number of positive and negative terms with automatically computed SO-PMI values, the performance is not always better. The results show improvement over using only GI for product reviews; however for movie reviews the accuracy decreases (especially for SO-PMI 1 which has too many negative terms). For Basic: GI the accuracy for movie reviews falls from 59.5% to 57.7% for Basic: GI & SO-PMI 1. This is probably due to the fact that

[3]We performed statistical significance tests using the paired t-test, as described in (Manning and Schütze, 1999), page 209. The data was randomly split in 6 sets.

the positive/negative labels computed with SO-PMI are not always reliable. It is not always the case that SO-PMI hurts the results though, for Basic GI & SO-PMI 2 the movie review accuracy improves to 63.2%. Movie reviews did better using GI & SO-PMI 2, and worse using GI & SO-PMI 1 than when just using GI, for both the Basic and Improved systems.

In most cases our method of classification performs better when classifying product reviews than movie reviews. This is not surprising, since movie reviews are known to be more difficult to classify (Turney, 2002; Turney and Littman, 2003). Movie reviews will often contain many sentences with objective information about the characters or the plot of the movie. Although these sentences are objective they may contain many positive and negative terms. This is even true of movie titles. Consider "The Good the Bad and the Ugly". It is a very positively reviewed movie[4], however its title contains only one positive and two negative terms, as such repeating the title of the film in the review would make the review seem more negative. Similar problems might exist for product reviews, maybe to a lesser extent. Other researchers have used earlier versions of the same movie review data set in their research; as such we can compare our best results with their best results. Support Vector Machines, in combination with a method of distinguishing subjective from objective statements was used on a previous version of this data set. The accuracy reported from this test was 86% (Pang et al., 2004). The best accuracy found using our method was 66.7% for the third last system in Table 1. On product reviews from epinions.com our best classification accuracy was 69.3%, for the improved system that uses GI & CTRW and for the one that uses GI & SO-PMI 1. We had seven different types of products, so the performance is in a comparable range with the average of 74% reported in (Turney, 2002) for four types of products.

Our improved system, in one of its best variants (Improved: GI & CTRW & Adj), achieved a statistically significant increase of 7.2 percentage points for movie reviews, compared to the baseline basic system (Basic: GI) that counts positive/negative terms from GI (from 59.5% to 66.7%). For product reviews the improvement over the baseline is smaller and not statistically significant. When comparing the baseline system (Basic: GI) with the best system (Improved: GI & SO-PMI 1) the results improve by 1.4 percentage points, from 67.9% to 69.3%.

6 Future Work

There are many possible directions for future work. For example, we would like to use only subjective sentences in classifying reviews. Methods of training classifiers to determine the objectivity of a sentence have been examined before (Pang et al., 2004). By using similar methods it could be possible to eliminate unwanted objective sentences, which may still contain positive or negative terms. An alternative way to improve the accuracy of classifying movie and product reviews could be to build small domain models of salient objective keyphrases. Positive and negative terms in these keyphrases would be ignored in the term counting method.

One of the weaknesses of our term counting method is that no word sense disambiguation is performed on the terms in the sentences. Although, in many cases if one sense of a term is positive, the other senses are also positive, this is not always true. Providing a word sense disambiguation method could improve the performance of the system. Also, parsing the texts to get the precise scope of negations and intensifiers should help; manual inspection can be used to correct any parsing errors related to these scopes.

Some positive and negative terms may not all be equally positive or negative. Positive and

[4] http://www.rottentomatoes.com/m/good_the_bad_and_the_ugly/

negative terms can be given weights to show just how positive or negative they are. Overstatements and understatements could also be weighted.

Acknowledgments

We wish to thank Egidio Terra and Charlie Clarke for giving us permission to use the Waterloo MultiText System with the terabyte corpus of web data, and Peter Turney and his colleagues at NRC/IIT for giving us access to their local copy of this system. Our research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the University of Ottawa.

References

Xue Bai, Rema Padman, and Edoardo Airoldi. 2004. Sentiment extraction from unstructured text using tabu search-enhanced markov blanket. In *Proceedings of the International Workshop on Mining for and from the Semantic Web.*

Sergio Bolasco and Francesca della Ratta-Rinaldi. 2004. Experiments on semantic categorization of texts: analysis of positive and negative dimension. In *Proceedings of the 7th International Conference on the Statistical Analysis of Textual Data (JADT 2004).*

Charles L. A. Clarke and Egidio Terra. 2003. Passage retrieval vs. document retrieval for factoid question answering. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Toronto, Canada.

Michael Gamon. 2004. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings the 20th International Conference on Computational Linguistics.*

Vasileios Hatzivassiloglou and Kathleen McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL'97)*, pages 174–181, Madrid, Spain.

S. I. Hayakawa, editor. 1994. *Choose the Right Word.* Second Edition, revised by Eugene Ehrlich. HarperCollins Publishers.

Diana Zaiu Inkpen, Olga Feiguina, and Graeme Hirst. 2004. Generating more-positive and more-negative text. In *Proceedings of the AAAI Symposium on Exploring Attitude and Affect in Text: Theories and Applications (published as AAAI technical report SS-04-07).*

Christopher Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing.* The MIT Press, Cambridge, Massachusetts.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, Philadelphia, PA.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain.

Livia Polanyi and Annie Zaenen. 2004. Contextual valence shifters. In *Proceedings of the AAAI Symposium on Exploring Attitude and Affect in Text: Theories and Applications (published as AAAI technical report SS-04-07).*

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program, 14(3): 130-137.*

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 105–112.

Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, Daniel M. Ogilvie, and associates. 1966. *The General Inquirer: A Computer Approach to Content Analysis.* The MIT Press.

Maite Taboada and Jack Grieve. 2004. Analyzing appraisal automatically. In *Proceedings of the AAAI Symposium on Exploring Attitude and Affect in Text: Theories and Applications.*

P.D. Turney and M.L. Littman. 2002. Unsupervised learning of semantic orientation from a hundred-billion-word corpus. Technical Report ERB-1094, National Research Council, Institute for Information Technology.

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

P.D. Turney. 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 2002)*, pages 417–424, Philadelphia, PA.

Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. 2004. Learning subjective language. *Computational Linguistics 30(3).*

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 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003).*