Examining the Role of Linguistic Knowledge Sources in the Automatic Identification and Classification of Reviews

Vincent Ng and **Sajib Dasgupta** and **S. M. Niaz Arifin**
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688
{vince,sajib,arif}@hlt.utdallas.edu

Abstract

This paper examines two problems in document-level sentiment analysis: (1) determining whether a given document is a review or not, and (2) classifying the polarity of a review as positive or negative. We first demonstrate that review identification can be performed with high accuracy using only unigrams as features. We then examine the role of four types of simple linguistic knowledge sources in a polarity classification system.

1 Introduction

Sentiment analysis involves the identification of positive and negative opinions from a text segment. The task has recently received a lot of attention, with applications ranging from multi-perspective question-answering (e.g., Cardie et al. (2004)) to opinion-oriented information extraction (e.g., Riloff et al. (2005)) and summarization (e.g., Hu and Liu (2004)). Research in sentiment analysis has generally proceeded at three levels, aiming to identify and classify opinions from *documents*, *sentences*, and *phrases*. This paper examines two problems in document-level sentiment analysis, focusing on analyzing a particular type of opinionated documents: *reviews*.

The first problem, *polarity classification*, has the goal of determining a review's polarity — *positive* ("thumbs up") or *negative* ("thumbs down"). Recent work has expanded the polarity classification task to additionally handle documents expressing a *neutral* sentiment. Although studied fairly extensively, polarity classification remains a challenge to natural language processing systems.

We will focus on an important linguistic aspect of polarity classification: examining the role of a variety of simple, yet under-investigated, linguistic knowledge sources in a learning-based polarity classification system. Specifically, we will show how to build a high-performing polarity classifier by exploiting information provided by (1) high order n-grams, (2) a lexicon composed of adjectives manually annotated with their polarity information (e.g., *happy* is annotated as positive and *terrible* as negative), (3) dependency relations derived from dependency parses, and (4) objective terms and phrases extracted from *neutral* documents.

As mentioned above, the majority of work on document-level sentiment analysis to date has focused on polarity classification, assuming as input a set of reviews to be classified. A relevant question is: what if we don't know that an input document is a review in the first place? The second task we will examine in this paper — *review identification* — attempts to address this question. Specifically, review identification seeks to determine whether a given document is a review or not.

We view both *review identification* and *polarity classification* as a classification task. For review identification, we train a classifier to distinguish movie reviews and movie-related non-reviews (e.g., movie ads, plot summaries) using only unigrams as features, obtaining an accuracy of over 99% via 10-fold cross-validation. Similar experiments using documents from the book domain also yield an accuracy as high as 97%. An analysis of the results reveals that the high accuracy can be attributed to the difference in the vocabulary employed in reviews and non-reviews: while reviews can be composed of a mixture of subjective and objective language, our non-review documents rarely contain subjective expressions.

Next, we learn our polarity classifier using positive and negative reviews taken from two movie review datasets, one assembled by Pang and Lee (2004) and the other by ourselves. The resulting classifier, when trained on a feature set derived from the four types of linguistic knowledge sources mentioned above, achieves a 10-fold cross-validation accuracy of 90.5% and 86.1% on Pang et al.'s dataset and ours, respectively. To our knowledge, our result on Pang et al.'s dataset is one of the best reported to date. Perhaps more importantly, an analysis of these results show that the various types of features interact in an interesting manner, allowing us to draw conclusions that provide new insights into polarity classification.

2 Related Work

2.1 Review Identification

As noted in the introduction, while a review can contain both subjective and objective phrases, our non-reviews are essentially factual documents in which subjective expressions can rarely be found. Hence, review identification can be viewed as an instance of the broader task of classifying whether a document is *mostly factual/objective* or *mostly opinionated/subjective*. There have been attempts on tackling this so-called *document-level subjectivity classification* task, with very encouraging results (see Yu and Hatzivassiloglou (2003) and Wiebe et al. (2004) for details).

2.2 Polarity Classification

There is a large body of work on classifying the polarity of a document (e.g., Pang et al. (2002), Turney (2002)), a sentence (e.g., Liu et al. (2003), Yu and Hatzivassiloglou (2003), Kim and Hovy (2004), Gamon et al. (2005)), a phrase (e.g., Wilson et al. (2005)), and a specific object (such as a product) mentioned in a document (e.g., Morinaga et al. (2002), Yi et al. (2003), Popescu and Etzioni (2005)). Below we will center our discussion of related work around the four types of features we will explore for polarity classification.

Higher-order n-grams. While n-grams offer a simple way of capturing context, previous work has rarely explored the use of n-grams as features in a polarity classification system beyond unigrams. Two notable exceptions are the work of Dave et al. (2003) and Pang et al. (2002). Interestingly, while Dave et al. report good performance on classifying reviews using bigrams or trigrams alone, Pang et al. show that bigrams are not useful features for the task, whether they are used in isolation or in conjunction with unigrams. This motivates us to take a closer look at the utility of higher-order n-grams in polarity classification.

Manually-tagged term polarity. Much work has been performed on learning to identify and classify *polarity* terms (i.e., terms expressing a positive sentiment (e.g., *happy*) or a negative sentiment (e.g., *terrible*)) and exploiting them to do polarity classification (e.g., Hatzivassiloglou and McKeown (1997), Turney (2002), Kim and Hovy (2004), Whitelaw et al. (2005), Esuli and Sebastiani (2005)). Though reasonably successful, these (semi-)automatic techniques often yield lexicons that have either high coverage/low precision or low coverage/high precision. While manually constructed positive and negative word lists exist (e.g., General Inquirer[1]), they too suffer from the problem of having low coverage. This prompts us to manually construct our own polarity word lists[2] and study their use in polarity classification.

Dependency relations. There have been several attempts at extracting features for polarity classification from dependency parses, but most focus on extracting specific types of information such as *adjective-noun relations* (e.g., Dave et al. (2003), Yi et al. (2003)) or *nouns* that enjoy a dependency relation with a polarity term (e.g., Popescu and Etzioni (2005)). Wilson et al. (2005) extract a larger variety of features from dependency parses, but unlike us, their goal is to determine the polarity of a *phrase*, not a document. In comparison to previous work, we investigate the use of a larger set of dependency relations for classifying reviews.

Objective information. The objective portions of a review do not contain the author's opinion; hence features extracted from objective sentences and phrases are irrelevant with respect to the polarity classification task and their presence may complicate the learning task. Indeed, recent work has shown that benefits can be made by first separating *facts* from *opinions* in a document (e.g, Yu and Hatzivassiloglou (2003)) and classifying the polarity based solely on the subjective portions of the document (e.g., Pang and Lee (2004)). Motivated by the work of Koppel and Schler (2005), we identify and extract objective material from *non-reviews* and show how to exploit such information in polarity classification.

[1] `http://www.wjh.harvard.edu/~inquirer/spreadsheet_guid.htm`

[2] Wilson et al. (2005) have also manually tagged a list of terms with their polarity, but this list is not publicly available.

Finally, previous work has also investigated features that do not fall into any of the above categories. For instance, instead of representing the polarity of a term using a binary value, Mullen and Collier (2004) use Turney's (2002) method to assign a real value to represent term polarity and introduce a variety of numerical features that are aggregate measures of the polarity values of terms selected from the document under consideration.

3 Review Identification

Recall that the goal of review identification is to determine whether a given document is a review or not. Given this definition, two immediate questions come to mind. First, should this problem be addressed in a domain-specific or domain-independent manner? In other words, should a review identification system take as input documents coming from the same domain or not?

Apparently this is a design question with no definite answer, but our decision is to perform domain-specific review identification. The reason is that the primary motivation of review identification is the need to identify reviews for further analysis by a polarity classification system. Since polarity classification has almost exclusively been addressed in a domain-specific fashion, it seems natural that its immediate upstream component — review identification — should also assume domain specificity. Note, however, that assuming domain specificity is not a self-imposed limitation. In fact, we envision that the review identification system will have as its upstream component a text classification system, which will classify documents by topic and pass to the review identifier only those documents that fall within its domain.

Given our choice of domain specificity, the next question is: which documents are non-reviews? Here, we adopt a simple and natural definition: a non-review is any document that belongs to the given domain but is not a review.

Dataset. Now, recall from the introduction that we cast review identification as a classification task. To train and test our review identifier, we use 2000 reviews and 2000 non-reviews from the movie domain. The 2000 reviews are taken from Pang et al.'s polarity dataset (version 2.0)[3], which consists of an equal number of positive and negative reviews. We collect the non-reviews for the movie domain from the Internet Movie Database website[4], randomly selecting any documents from this site that are on the movie topic but are not reviews themselves. With this criterion in mind, the 2000 non-review documents we end up with are either movie ads or plot summaries.

Training and testing the review identifier. We perform 10-fold cross-validation (CV) experiments on the above dataset, using Joachims' (1999) SVM^{light} package[5] to train an SVM classifier for distinguishing reviews and non-reviews. All learning parameters are set to their default values.[6] Each document is first tokenized and downcased, and then represented as a vector of unigrams with length normalization.[7] Following Pang et al. (2002), we use frequency as presence. In other words, the ith element of the document vector is 1 if the corresponding unigram is present in the document and 0 otherwise. The resulting classifier achieves an accuracy of 99.8%.

Classifying neutral reviews and non-reviews. Admittedly, the high accuracy achieved using such a simple set of features is somewhat surprising, although it is consistent with previous results on document-level subjectivity classification in which accuracies of 94-97% were obtained (Yu and Hatzivassiloglou, 2003; Wiebe et al., 2004). Before concluding that review classification is an easy task, we conduct an additional experiment: we train a review identifier on a new dataset where we keep the same 2000 non-reviews but replace the positive/negative reviews with 2000 *neutral* reviews (i.e., reviews with a mediocre rating). Intuitively, a neutral review contains fewer terms with strong polarity than a positive/negative review. Hence, this additional experiment would allow us to investigate whether the lack of strong polarized terms in neutral reviews would increase the difficulty of the learning task.

Our neutral reviews are randomly chosen from Pang et al.'s pool of 27886 unprocessed movie reviews[8] that have either a rating of 2 (on a 4-point scale) or 2.5 (on a 5-point scale). Each review then undergoes a semi-automatic preprocessing stage

[3] Available from `http://www.cs.cornell.edu/people/pabo/movie-review-data`.

[4] See `http://www.imdb.com`.

[5] Available from `svmlight.joachims.org`.

[6] We tried polynomial and RBF kernels, but none yields better performance than the default linear kernel.

[7] We observed that not performing length normalization hurts performance slightly.

[8] Also available from Pang's website. See Footnote 3.

where (1) HTML tags and any header and trailer information (such as date and author identity) are removed; (2) the document is tokenized and downcased; (3) the rating information extracted by regular expressions is removed; and (4) the document is manually checked to ensure that the rating information is successfully removed. When trained on this new dataset, the review identifier also achieves an accuracy of 99.8%, suggesting that this learning task isn't any harder in comparison to the previous one.

Discussion. We hypothesized that the high accuracies are attributable to the different vocabulary used in reviews and non-reviews. As part of our verification of this hypothesis, we plot the learning curve for each of the above experiments.[9] We observe that a 99% accuracy was achieved in all cases even when only 200 training instances are used to acquire the review identifier. The ability to separate the two classes with such a small amount of training data seems to imply that features strongly indicative of one or both classes are present. To test this hypothesis, we examine the "informative" features for both classes. To get these informative features, we rank the features by their weighted log-likelihood ratio (WLLR)[10]:

$$P(w_t|c_j)\log\frac{P(w_t|c_j)}{P(w_t|\neg c_j)},$$

where w_t and c_j denote the tth word in the vocabulary and the jth class, respectively. Informally, a feature (in our case a unigram) w will have a high rank with respect to a class c if it appears frequently in c and infrequently in other classes. This correlates reasonably well with what we think an informative feature should be. A closer examination of the feature lists sorted by WLLR confirms our hypothesis that each of the two classes has its own set of distinguishing features.

Experiments with the book domain. To understand whether these good review identification results only hold true for the movie domain, we conduct similar experiments with book reviews and non-reviews. Specifically, we collect 1000 book reviews (consisting of a mixture of positive, negative, and neutral reviews) from the Barnes and Noble website[11], and 1000 non-reviews that are on the book topic (mostly book summaries) from Amazon.[12] We then perform 10-fold CV experiments using these 2000 documents as before, achieving a high accuracy of 96.8%. These results seem to suggest that automatic review identification can be achieved with high accuracy.

4 Polarity Classification

Compared to review identification, polarity classification appears to be a much harder task. This section examines the role of various linguistic knowledge sources in our learning-based polarity classification system.

4.1 Experimental Setup

Like several previous work (e.g., Mullen and Collier (2004), Pang and Lee (2004), Whitelaw et al. (2005)), we view polarity classification as a supervised learning task. As in review identification, we use SVM^{light} with default parameter settings to train polarity classifiers[13], reporting all results as 10-fold CV accuracy.

We evaluate our polarity classifiers on two movie review datasets, each of which consists of 1000 positive reviews and 1000 negative reviews. The first one, which we will refer to as Dataset A, is the Pang et al. polarity dataset (version 2.0). The second one (Dataset B) was created by us, with the sole purpose of providing additional experimental results. Reviews in Dataset B were randomly chosen from Pang et al.'s pool of 27886 unprocessed movie reviews (see Section 3) that have either a positive or a negative rating. We followed exactly Pang et al.'s guideline when determining whether a review is positive or negative.[14] Also, we took care to ensure that reviews included in Dataset B do not appear in Dataset A. We applied to these reviews the same four pre-processing steps that we did to the neutral reviews in the previous section.

4.2 Results

The baseline classifier. We can now train our baseline polarity classifier on each of the two

[9]The curves are not shown due to space limitations.

[10]Nigam et al. (2000) show that this metric is effective at selecting good features for text classification. Other commonly-used feature selection metrics are discussed in Yang and Pedersen (1997).

[11]`www.barnesandnoble.com`

[12]`www.amazon.com`

[13]We also experimented with polynomial and RBF kernels when training polarity classifiers, but neither yields better results than linear kernels.

[14]The guidelines come with their polarity dataset. Briefly, a positive review has a rating of $\geq$ 3.5 (out of 5) or $\geq$ 3 (out of 4), whereas a negative review has a rating of $\leq$ 2 (out of 5) or $\leq$ 1.5 (out of 4).

System Variation	Dataset A	Dataset B
Baseline	87.1	82.7
Adding bigrams and trigrams	89.2	84.7
Adding dependency relations	89.0	84.5
Adding polarity info of adjectives	90.4	**86.2**
Discarding objective materials	**90.5**	86.1

Table 1: Polarity classification accuracies.

datasets. Our baseline classifier employs as features the k highest-ranking unigrams according to WLLR, with k/2 features selected from each class. Results with k = 10000 are shown in row 1 of Table 1.[15] As we can see, the baseline achieves an accuracy of 87.1% and 82.7% on Datasets A and B, respectively. Note that our result on Dataset A is as strong as that obtained by Pang and Lee (2004) via their subjectivity summarization algorithm, which retains only the subjective portions of a document.

As a sanity check, we duplicated Pang et al.'s (2002) baseline in which all unigrams that appear four or more times in the training documents are used as features. The resulting classifier achieves an accuracy of 87.2% and 82.7% for Datasets A and B, respectively. Neither of these results are significantly different from our baseline results.[16]

Adding higher-order n-grams. The negative results that Pang et al. (2002) obtained when using bigrams as features for their polarity classifier seem to suggest that high-order n-grams are not useful for polarity classification. However, recent research in the related (but arguably simpler) task of text classification shows that a bigram-based text classifier outperforms its unigram-based counterpart (Peng et al., 2003). This prompts us to re-examine the utility of high-order n-grams in polarity classification.

In our experiments we consider adding bigrams and trigrams to our baseline feature set. However, since these higher-order n-grams significantly outnumber the unigrams, adding all of them to the feature set will dramatically increase the dimensionality of the feature space and may undermine the impact of the unigrams in the resulting classifier. To avoid this potential problem, we keep the number of unigrams and higher-order n-grams equal. Specifically, we augment the baseline feature set (consisting of 10000 unigrams) with 5000 bigrams and 5000 trigrams. The bigrams and trigrams are selected based on their WLLR computed over the positive reviews and negative reviews in the training set for each CV run.

Results using this augmented feature set are shown in row 2 of Table 1. We see that accuracy rises significantly from 87.1% to 89.2% for Dataset A and from 82.7% to 84.7% for Dataset B. This provides evidence that polarity classification can indeed benefit from higher-order n-grams.

Adding dependency relations. While bigrams and trigrams are good at capturing local dependencies, dependency relations can be used to capture non-local dependencies among the constituents of a sentence. Hence, we hypothesized that our n-gram-based polarity classifier would benefit from the addition of dependency-based features.

Unlike most previous work on polarity classification, which has largely focused on exploiting adjective-noun (AN) relations (e.g., Dave et al. (2003), Popescu and Etzioni (2005)), we hypothesized that subject-verb (SV) and verb-object (VO) relations would also be useful for the task. The following (one-sentence) review illustrates why.

While I really like the actors, the plot is rather uninteresting.

A unigram-based polarity classifier could be confused by the simultaneous presence of the positive term *like* and the negative term *uninteresting* when classifying this review. However, incorporating the VO relation (*like*, *actors*) as a feature may allow the learner to learn that the author likes the actors and not necessarily the movie.

In our experiments, the SV, VO and AN relations are extracted from each document by the MINIPAR dependency parser (Lin, 1998). As with n-grams, instead of using all the SV, VO and AN relations as features, we select among them the best 5000 according to their WLLR and retrain the polarity classifier with our n-gram-based feature set augmented by these 5000 dependency-based features. Results in row 3 of Table 1 are somewhat surprising: the addition of dependency-based features does not offer any improvements over the simple n-gram-based classifier.

[15] We experimented with several values of k and obtained the best result with k = 10000.

[16] We use two-tailed paired t-tests when performing significance testing, with p set to 0.05 unless otherwise stated.

Incorporating manually tagged term polarity. Next, we consider incorporating a set of features that are computed based on the polarity of adjectives. As noted before, we desire a high-precision, high-coverage lexicon. So, instead of exploiting a learned lexicon, we manually develop one.

To construct the lexicon, we take Pang et al.'s pool of unprocessed documents (see Section 3), remove those that appear in either Dataset A or Dataset B[17], and compile a list of adjectives from the remaining documents. Then, based on heuristics proposed in psycholinguistics[18], we hand-annotate each adjective with its *prior polarity* (i.e., polarity in the absence of context). Out of the 45592 adjectives we collected, 3599 were labeled as positive, 3204 as negative, and 38789 as neutral. A closer look at these adjectives reveals that they are by no means domain-dependent despite the fact that they were taken from movie reviews.

Now let us consider a simple procedure P for deriving a feature set that incorporates information from our lexicon: (1) collect all the bigrams from the training set; (2) for each bigram that contains at least one adjective labeled as positive or negative according to our lexicon, create a new feature that is identical to the bigram except that each adjective is replaced with its polarity label[19]; (3) merge the list of newly generated features with the list of bigrams[20] and select the top 5000 features from the merged list according to their WLLR.

We then repeat procedure P for the trigrams and also the dependency features, resulting in a total of 15000 features. Our new feature set comprises these 15000 features as well as the 10000 unigrams we used in the previous experiments.

Results of the polarity classifier that incorporates term polarity information are encouraging (see row 4 of Table 1). In comparison to the classifier that uses only n-grams and dependency-based features (row 3), accuracy increases significantly (p = .1) from 89.2% to 90.4% for Dataset A, and from 84.7% to 86.2% for Dataset B. These results suggest that the classifier has benefited from the use of features that are less sparse than n-grams.

Using objective information. Some of the 25000 features we generated above correspond to n-grams or dependency relations that do not contain subjective information. We hypothesized that not employing these "objective" features in the feature set would improve system performance. More specifically, our goal is to use procedure P again to generate 25000 "subjective" features by ensuring that the objective ones are not selected for incorporation into our feature set.

To achieve this goal, we first use the following rote-learning procedure to identify objective material: (1) extract all unigrams that appear in objective documents, which in our case are the 2000 non-reviews used in review identification [see Section 3]; (2) from these "objective" unigrams, we take the best 20000 according to their WLLR computed over the non-reviews and the reviews in the training set for each CV run; (3) repeat steps 1 and 2 separately for bigrams, trigrams and dependency relations; (4) merge these four lists to create our 80000-element list of objective material.

Now, we can employ procedure P to get a list of 25000 "subjective" features by ensuring that those that appear in our 80000-element list are not selected for incorporation into our feature set.

Results of our classifier trained using these subjective features are shown in row 5 of Table 1. Somewhat surprisingly, in comparison to row 4, we see that our method for filtering objective features does not help improve performance on the two datasets. We will examine the reasons in the following subsection.

4.3 Discussion and Further Analysis

Using the four types of knowledge sources previously described, our polarity classifier significantly outperforms a unigram-based baseline classifier. In this subsection, we analyze some of these results and conduct additional experiments in an attempt to gain further insight into the polarity classification task. Due to space limitations, we will simply present results on Dataset A below, and show results on Dataset B only in cases where a different trend is observed.

The role of feature selection. In all of our experiments we used the best k features obtained via WLLR. An interesting question is: how will these results change if we do not perform feature selection? To investigate this question, we conduct two

[17]We treat the test documents as unseen data that should not be accessed for any purpose during system development.

[18]`http://www.sci.sdsu.edu/CAL/wordlist`

[19]Neutral adjectives are not replaced.

[20]A newly generated feature could be misleading for the learner if the *contextual polarity* (i.e., polarity in the presence of context) of the adjective involved differs from its prior polarity (see Wilson et al. (2005)). The motivation behind merging with the bigrams is to create a feature set that is more robust in the face of potentially misleading generalizations.

experiments. First, we train a polarity classifier using all unigrams from the training set. Second, we train another polarity classifier using all unigrams, bigrams, and trigrams. We obtain an accuracy of 87.2% and 79.5% for the first and second experiments, respectively.

In comparison to our baseline classifier, which achieves an accuracy of 87.1%, we can see that using all unigrams does not hurt performance, but performance drops abruptly with the addition of all bigrams and trigrams. These results suggest that feature selection is critical when bigrams and trigrams are used in conjunction with unigrams for training a polarity classifier.

The role of bigrams and trigrams. So far we have seen that training a polarity classifier using only unigrams gives us reasonably good, though not outstanding, results. Our question, then, is: would bigrams alone do a better job at capturing the sentiment of a document than unigrams? To answer this question, we train a classifier using all bigrams (without feature selection) and obtain an accuracy of 83.6%, which is significantly worse than that of a unigram-only classifier. Similar results were also obtained by Pang et al. (2002).

It is possible that the worse result is due to the presence of a large number of irrelevant bigrams. To test this hypothesis, we repeat the above experiment except that we only use the best 10000 bigrams selected according to WLLR. Interestingly, the resulting classifier gives us a lower accuracy of 82.3%, suggesting that the poor accuracy is not due to the presence of irrelevant bigrams.

To understand why using bigrams alone does not yield a good classification model, we examine a number of test documents and find that the feature vectors corresponding to some of these documents (particularly the short ones) have all zeroes in them. In other words, none of the bigrams from the training set appears in these reviews. This suggests that the main problem with the bigram model is likely to be data sparseness. Additional experiments show that the trigram-only classifier yields even worse results than the bigram-only classifier, probably because of the same reason.

Nevertheless, these higher-order n-grams play a non-trivial role in polarity classification: we have shown that the addition of bigrams and trigrams selected via WLLR to a unigram-based classifier significantly improves its performance.

The role of dependency relations. In the previous subsection we see that dependency relations do not contribute to overall performance on top of bigrams and trigrams. There are two plausible reasons. First, dependency relations are simply not useful for polarity classification. Second, the higher-order n-grams and the dependency-based features capture essentially the same information and so using either of them would be sufficient.

To test the first hypothesis, we train a classifier using only 10000 unigrams and 10000 dependency-based features (both selected according to WLLR). For Dataset A, the classifier achieves an accuracy of 87.1%, which is statistically indistinguishable from our baseline result. On the other hand, the accuracy for Dataset B is 83.5%, which is significantly better than the corresponding baseline (82.7%) at the p = .1 level. These results indicate that dependency information is somewhat useful for the task when bigrams and trigrams are not used. So the first hypothesis is not entirely true.

So, it seems to be the case that the dependency relations do not provide useful knowledge for polarity classification only in the presence of bigrams and trigrams. This is somewhat surprising, since these n-grams do not capture the non-local dependencies (such as those that may be present in certain SV or VO relations) that should intuitively be useful for polarity classification.

To better understand this issue, we again examine a number of test documents. Our initial investigation suggests that the problem might have stemmed from the fact that MINIPAR returns dependency relations in which all the verb inflections are removed. For instance, given the sentence *My cousin Paul really likes this long movie*, MINIPAR will return the VO relation ($like$, $movie$). To see why this can be a problem, consider another sentence *I like this long movie*. From this sentence, MINIPAR will also extract the VO relation ($like$, $movie$). Hence, this same VO relation is capturing two different situations, one in which the author himself likes the movie, and in the other, the author's cousin likes the movie. The overgeneralization resulting from these "stemmed" relations renders dependency information not useful for polarity classification. Additional experiments are needed to determine the role of dependency relations when stemming in MINIPAR is disabled.

The role of objective information. Results from the previous subsection suggest that our method for extracting objective materials and removing them from the reviews is not effective in terms of improving performance. To determine the reason, we examine the n-grams and the dependency relations that are extracted from the non-reviews. We find that only in a few cases do these extracted objective materials appear in our set of 25000 features obtained in Section 4.2. This explains why our method is not as effective as we originally thought. We conjecture that more sophisticated methods would be needed in order to take advantage of objective information in polarity classification (e.g., Koppel and Schler (2005)).

5 Conclusions

We have examined two problems in document-level sentiment analysis, namely, review identification and polarity classification. We first found that review identification can be achieved with very high accuracies (97-99%) simply by training an SVM classifier using unigrams as features. We then examined the role of several linguistic knowledge sources in polarity classification. Our results suggested that bigrams and trigrams selected according to the weighted log-likelihood ratio as well as manually tagged term polarity information are very useful features for the task. On the other hand, no further performance gains are obtained by incorporating dependency-based information or filtering objective materials from the reviews using our proposed method. Nevertheless, the resulting polarity classifier compares favorably to state-of-the-art sentiment classification systems.

References

C. Cardie, J. Wiebe, T. Wilson, and D. Litman. 2004. Low-level annotations and summary representations of opinions for multi-perspective question answering. In *New Directions in Question Answering*. AAAI Press/MIT Press.

K. Dave, S. Lawrence, and D. M. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proc. of WWW*, pages 519–528.

A. Esuli and F. Sebastiani. 2005. Determining the semantic orientation of terms through gloss classification. In *Proc. of CIKM*, pages 617–624.

M. Gamon, A. Aue, S. Corston-Oliver, and E. K. Ringger. 2005. Pulse: Mining customer opinions from free text. In *Proc. of the 6th International Symposium on Intelligent Data Analysis*, pages 121–132.

V. Hatzivassiloglou and K. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proc. of the ACL/EACL*, pages 174–181.

M. Hu and B. Liu. 2004. Mining and summarizing customer reviews. In *Proc. of KDD*, pages 168–177.

T. Joachims. 1999. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*, pages 44–56. MIT Press.

S.-M. Kim and E. Hovy. 2004. Determining the sentiment of opinions. In *Proc. of COLING*, pages 1367–1373.

M. Koppel and J. Schler. 2005. Using neutral examples for learning polarity. In *Proc. of IJCAI (poster)*.

D. Lin. 1998. Dependency-based evaluation of MINIPAR. In *Proc. of the LREC Workshop on the Evaluation of Parsing Systems*, pages 48–56.

H. Liu, H. Lieberman, and T. Selker. 2003. A model of textual affect sensing using real-world knowledge. In *Proc. of Intelligent User Interfaces (IUI)*, pages 125–132.

S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. 2002. Mining product reputations on the web. In *Proc. of KDD*, pages 341–349.

T. Mullen and N. Collier. 2004. Sentiment analysis using support vector machines with diverse information sources. In *Proc. of EMNLP*, pages 412–418.

K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134.

B. Pang and L. Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. of the ACL*, pages 271–278.

B. Pang, L. Lee, and S. Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proc. of EMNLP*, pages 79–86.

F. Peng, D. Schuurmans, and S. Wang. 2003. Language and task independent text categorization with simple language models. In *HLT/NAACL: Main Proc.* , pages 189–196.

A.-M. Popescu and O. Etzioni. 2005. Extracting product features and opinions from reviews. In *Proc. of HLT-EMNLP*, pages 339–346.

E. Riloff, J. Wiebe, and W. Phillips. 2005. Exploiting subjectivity classification to improve information extraction. In *Proc. of AAAI*, pages 1106–1111.

P. Turney. 2002. Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proc. of the ACL*, pages 417–424.

C. Whitelaw, N. Garg, and S. Argamon. 2005. Using appraisal groups for sentiment analysis. In *Proc. of CIKM*, pages 625–631.

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

T. Wilson, J. M. Wiebe, and P. Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proc. of EMNLP*, pages 347–354.

Y. Yang and J. O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Proc. of ICML*, pages 412–420.

J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack. 2003. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proc. of the IEEE International Conference on Data Mining (ICDM)*.

H. Yu and V. Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proc. of EMNLP*, pages 129–136.