Feature Selection and Weighting Methods in Sentiment Analysis

Tim O'Keefe
School of Information Technologies
University of Sydney
NSW 2006, Australia
toke9145@uni.sydney.edu.au

Irena Koprinska
School of Information Technologies
University of Sydney
NSW 2006, Australia
irena@it.usyd.edu.au

Abstract *Sentiment analysis is the task of identifying whether the opinion expressed in a document is positive or negative about a given topic. Unfortunately, many of the potential applications of sentiment analysis are currently infeasible due to the huge number of features found in standard corpora. In this paper we systematically evaluate a range of feature selectors and feature weights with both Naïve Bayes and Support Vector Machine classifiers. This includes the introduction of two new feature selection methods and three new feature weighting methods. Our results show that it is possible to maintain a state-of-the art classification accuracy of 87.15% while using less than 36% of the features.*

Keywords Information Retrieval, Natural Language Techniques and Documents

1 Introduction

The opinions of other people have always been important to us, and in particular we are often concerned with the prevailing sentiment of those opinions. Often governments want to know how voters feel about a policy, corporations want to know how customers feel about a product and movie goers want to know if others would recommend a movie. The idea behind sentiment analysis is to provide this information by building a system that can classify documents as positive or negative, according to the overall sentiment expressed within those documents.

Early approaches to sentiment analysis tended to focus on classifying documents according to the out-of-context sentiment of individual features [14]. While these approaches did not require domain-specific training data, their accuracy was quite poor. Subsequent research focused on supervised learning techniques that are common in text categorisation tasks [9], such as Support Vector Machine (SVM) and Naïve Bayes (NB) classifiers. Though these techniques are far more accurate than the earlier text-based approaches, they are a lot more computationally expensive to run due to the large number of features.

Proceedings of the 14th Australasian Document Computing Symposium, Sydney, Australia, 4 December 2009.

In fact, in the Pang et al. [9] movie review data set that has become the *de facto* standard there are just under 51,000 unique words and symbols. Very few of these features actually provide useful information to the classifier, so feature selection can be used to reduce the number of features. Despite the fact that its use is commonplace, there has been little research into the effects of different methods of feature selection in sentiment analysis. In this paper we address this gap by comparing three feature selection methods at a number of selection thresholds, using six feature weighting methods. The feature selection methods include Categorical Proportional Difference (PD), a recently proposed method that was successfully used for topic-based text categorisation, and two methods based on sentiment values from SentiWordNet (SWN) [2] that we introduce: SWNSS and SWNPD. The feature weighting methods include Feature Frequency (FF), Feature Presence (FP), TFIDF, and three other methods based on words grouped by their SWN values that we introduce: SWN-SG, SWN-PG and SWN-PS. All tests were conducted using both SVM and NB.

Our results show that PD and SWNSS were able to maintain or improve accuracy when used with suitable weightings while SWNPD tended to reduce accuracy, though not in all cases. SVM with PD as a feature selector achieved our highest accuracy of 87.15% which is comparable with the state-of-the art, but uses a vastly reduced set of features.

2 Background

While there was some early work in word-level sentiment analysis [3] and a semi-automatic approach to document-level sentiment analysis [13], the real genesis of document-level sentiment analysis was the work of Turney [14]. The basic idea behind Turney's approach was to average the sentiment of the adjectives within each document and then classify the document depending on whether the average was positive or negative. To find the sentiment of adjectives, Turney used the AltaVista search engine to determine how often individual adjectives co-occured with the words "excellent" and "poor." Words that co-occured more often with "excellent" were deemed positive and words co-occuring more often with "poor" were deemed negative.

Authors	Data split	Classifier	Cross Validation	Feature Selection	Baseline Accuracy (%)	Best Accuracy (%)
Pang et al. [9]	700+ 700-	NB, ME, SVM	3-fold	No	N/A	82.9
Pang & Lee [8]	1000+ 1000-	NB, SVM	10-fold	Yes	87.15	87.2
Mullen & Collier [7]	700+ 700-	Hybrid SVM (Turney values, Osgood values, lemma models)	10-fold	No	83.5	86
König & Brill [6]	1000+ 1000-	Pattern-based, SVM, Hybrid	5-fold	No	87.5	91
Abbasi et al. [1]	1000+ 1000-	Genetic Algorithms (GA), Information Gain (IG), IG + GA	10-fold	Yes	87.95	91.7
Prabowo & Thelwall [10]	1000+ 1000-	Hybrid (rule + closeness measure + SVM)	10-fold	No	87.3	87.3

Table 1: Results reported in the literature on various versions of the Pang et al. [9] movie review data set.

The first use of supervised learning in sentiment analysis was by Pang et al. [9]. Their aim was to determine whether sentiment analysis could be treated as a special case of topic-based categorisation with two topics: *positive* and *negative*. To achieve this they tested Naïve Bayes (NB), Maximum Entropy (ME), and Support Vector Machine (SVM) classifiers, all of which have performed well in topic-based categorisation. For features, they used the words and symbols of the documents as either a unigram or a bigram bag-of-features, with unigrams generally performing better. They tested Feature Frequency (FF) and Feature Presence (FP) and found that by using a SVM with unigram FP they could achieve an accuracy of 82.9% in a 3-fold cross validation test. Table 1 lists some of the best results that have been reported in the literature.

2.1 Feature Selection

Most researchers employ basic feature selection in their work in order to improve computational performance, with a few using more complicated approaches [5, 8, 1]. To date there have only been two papers that have entirely focused on using feature selection to improve sentiment analysis. The first was by Pang & Lee [8], who used a SVM trained on subjective and objective text to remove objective sentences from the corpus. In their initial results they found that document sentiment classification accuracy actually declined. They then conducted some "non-obvious feature engineering" by making it more likely that sentences adjacent to removed sentences would be removed as well, which slightly improved accuracy over their baseline.

The other work that used sophisticated feature selection was by Abbasi et al. [1]. They found that using either information gain (IG) or genetic algorithms (GA) resulted in an improvement in accuracy. They also combined the two in a new algorithm called the Entropy Weighted Genetic Algorithm (EWGA), which achieved the highest level of accuracy in sentiment analysis to date of 91.7%. The drawback of this new method is that while it can efficiently classify items, it is very computationally expensive to conduct the initial feature selection, since both GA and IG are expensive to run.

2.2 SentiWordNet

SentiWordNet (SWN) is an extension of WordNet that was developed by Esuli & Sebastiani [2], which is intended to augment the information in WordNet with information about the sentiment of the words in WordNet. Our research uses the information provided by sentiment in some detail, so we will describe it here. Each synset in SWN has a positive sentiment score, a negative sentiment score and an objectivity score. When these three scores are summed they equal one, so they give an indication of the relative strength of the positivity, negativity and objectivity of each synset. Esuli & Sebastiani [2] obtained these values by using several semi-supervised ternary classifiers, all of which were capable of determining whether a word was positive, negative, or objective. If all the classifiers agreed on a classification then the maximum value was assigned for the associated score, otherwise the values for the positive, negative and objective scores were proportional to the number of classifiers that assigned the word to each class.

The drawback in using SWN is that it requires word sense disambiguation to find the correct sense of a word and its associated scores. Whilst there has been significant research into this problem, we decided that it was out of scope to use any sophisticated word sense disambiguation for this project, so we simply took the highest positive and negative values that we could find for each word. This is based on the assumption that in a subjective document it is reasonably likely that the most subjective sense of a word is being used. Preliminary testing confirmed that using the most subjective senses tended to outperform the senses that are known to be most frequent.

3 Data & Evaluation

We use two different supervised learning approaches to sentiment analysis: Support Vector Machines (SVM) and Naïve Bayes (NB). SVM and NB classifiers were originally used in sentiment analysis by Pang et al. [9], who found that SVM classifiers generally outperformed NB. In order to be as comparable to Pang & Lee as possible we use the SVM implementation developed by Joachims [4], called SVM$_{LIGHT}$. For Naïve Bayes we use the implementation available in Weka [15].

The data set we use is the set of 1000 positive and 1000 negative movie reviews from IMDb[1] that was introduced in Pang et al. [9]. For all of our experiments we conduct 10-fold cross validation, and we use paired t-tests at a confidence level of 0.05 to establish significance.

4 Feature Weighting Methods

4.1 Unigram Features

In the domain of sentiment analysis, and more generally text categorisation, it is common to use the words and symbols within the corpus as features in the feature vectors. Though there are other ways of representing the words and symbols, we will be using unigrams, where each unique word or symbol is counted as one feature. Pang et al. [9] found that unigrams fairly comprehensively out-performed bigrams and combinations of unigrams and bigrams. The different feature weights for the unigrams are discussed below.

4.1.1 Feature Frequency (FF)

The simplest way to represent a document with a vector is the feature frequency method that was originally used in sentiment analysis by Pang et al. [9]. The method uses the term frequency, i.e. the frequency that each unigram occurs within a document, as the feature values for that document. So if the word "excellent" appeared in a document ten times, the associated feature would have a value of ten.

4.1.2 Feature Presence (FP)

Pang et al. [9] were also the first to use feature presence in sentiment analysis. Feature presence is very similar to feature frequency, except that rather than using the frequency of a unigram as its value, we would merely use a one, to indicate that the unigram exists in the document. Multiple occurrences of the same unigram are ignored, so we get a vector of binary values, with ones for each unique unigram that occurs in the document, and zeros for all unigrams that appear in the corpus but not in the document.

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

4.1.3 Term Frequency - Inverse Document Frequency (TF-IDF)

TF-IDF is a common metric used in text categorisation tasks [11], but its use in sentiment analysis has been less widespread, and surprisingly it does not appear to have been used as a unigram feature weight. TF-IDF is composed of two scores, term frequency and inverse document frequency. Term frequency is found by simply counting the number of times that a given term has occured in a given document, and inverse document frequency is found by dividing the total number of documents by the number of documents that a given word appears in. When these values are multiplied together we get a score that is highest for words that appear frequently in a few documents, and low for terms that appear frequently in every document, allowing us to find terms that are important in a document.

4.2 SentiWordNet Word Groups

While unigram features have emerged as the most accurate approach to sentiment analysis, there has still been significant work in using other types of features [14, 7, 10]. While most of this previous research has shown that grouping or summing words based on their out-of-context sentiment has not performed well on its own [14, 9], some researchers have used these sorts of features to augment unigrams [7]. We add to this research by using SWN to put the words found in each document into groups, which we can then use as features for classifiers.

4.2.1 SWN Word Score Groups (SWN-SG)

One of the interesting features of SWN is that there are only a limited number of values that the positive and negative word scores can take on, due to the way those scores are calculated. We can take advantage of this fact to group words with the same positive or negative score, so that rather than having features that correspond to words, we have features that correspond to groups of words. The value of a feature would then be the number of words in the document that have the same positive or negative SWN score. So for example if the sentence "The acting was excellent, the special effects were amazing, and the script was terrific" appeared in a document we might find that "excellent," "amazing," and "terrific" all had the same positive score. When we turn that sentence into a feature vector one of the features would correspond to that positive score and would have a value of three, since there are three words with that score.

4.2.2 SWN Word Polarity Groups (SWN-PG)

Since SWN gives words both a positive and negative score, we can find whether a word is more positive than negative and vice versa. This allows us to define two features, positive and negative, which correspond to the counts of positive and negative words respectively. So words that are more positive than negative add one to the positive feature and words that are more negative add one to the negative feature. The end result is a feature vector with two features, the first being the number of positive words and the second being the number of negative words in the document.

4.2.3 SWN Word Polarity Sums (SWN-PS)

The final feature type that we introduce is similar to the word polarity groups, except that we actually sum the positive and negative scores, rather than just tallying the number of words with those scores. So when we convert a document into a feature vector there are two features. The first one is the sum of the SWN positive scores of all words that have a higher positive than negative score. The second feature is the sum of the SWN negative scores of all words that have a higher negative score than positive score. Any words that have no positive and no negative score, or where the positive and negative scores are equal, are ignored. The scores are adjusted for document length, so different length documents can be more accurately compared.

5 Feature Selection

When we set out to classify a document we generally start off with a very large number of words that need to be considered, even though very few of the words in the corpus are actually expressing sentiment. These extra features have two clear drawbacks that we would like to eliminate. The first is that they make document classification slower, since there are far more words than there really needs to be. The second is that they can actually reduce accuracy, since the classifier must consider these words when classifying a document.

Clearly there is an advantage in using fewer features, so in order to remove some of the unnecessary features, we use *feature selection*. As the name suggests, feature selection is a process where we run through the corpus before the classifier has been trained and remove any features that seem unnecessary. This allows the classifier to fit a model to the problem set more quickly since there is less information to consider, and thus allows it to classify items faster. In this section we describe several different methods of feature selection.

5.1 Categorical Proportional Difference (PD)

Categorical Proportional Difference (PD), introduced by Simeon & Hilderman [12], is a metric which tells us how close to being equal two numbers are. We can use this to find unigrams that occur mostly in one class of documents or the other, by using the positive document frequency and negative document frequency of a unigram as the two numbers. In other words if a unigram occurs predominantly in positive documents or predominantly in negative documents then the PD of the unigram will be close to one, whereas if it occurs in about as many positive documents as negative documents then its PD will be close to zero. While Simeon & Hilderman use a more general equation for multi-class problems, we use a simplified equation for our two-class problem, which is as follows:

$$\frac{|PositiveDF - NegativeDF|}{PositiveDF + NegativeDF}$$

A high score from this equation indicates that the unigram is telling us a lot, and a low score indicates that the unigram is telling us very little. For example if the word “actor” appears in exactly as many positive documents as negative documents then finding the word “actor” in a new document will tell us nothing about it and as such its PD score will be zero. Conversely, if the word “excellent” appears in only positive documents then finding the word “excellent” in a new document would give us a good clue that the document is positive, and as such it would have a PD score of one. So to use PD as a feature selector we simply need to remove any features where the result of the equation is less than or equal to some threshold value.

5.2 SWN Subjectivity Scores (SWNSS)

The SWN feature selector is actually able to distinguish objective and subjective terms, which is useful since only subjective terms should carry sentiment. To do this we use the SWN *subjectivity score*, which is found by adding the positive and negative SWN scores of a unigram together. This is the opposite of the *objectivity score* that is defined by Esuli & Sebastiani [2], but its use is equivalent. To use it as a feature selector we simply remove any unigrams whose subjective score is less than a certain threshold. When this feature selector is used, unigrams that are not found in SWN, such as names and misspellings, are removed from the corpus as well (although arguably the names of certain actors could give strong clues about the quality of a movie).

5.3 SWN Proportional Difference (SWNPD)

While the SWN subjectivity feature selector can find words that have some *a priori* sentiment attached, it cannot tell us whether that sentiment is consistent or meaningful. It is entirely possible that a word may have a SWN subjectivity score of one, indicating that it is very subjective, but its positive and negative scores may be 0.5 each. This may make the word uninformative as a feature so there could be value in removing it. To do this we define SWN Proportional Difference, which uses the SWN positive and negative scores in the PD equation, as follows:

$$\frac{|SWNPos - SWNNeg|}{SWNPos + SWNNeg}$$

Similarly to PD, SWNPD will be high for words that are mostly positive or negative, and low for words that are a mix of both. By using this score we hope to remove subjective words that have an ambiguous polarity from the corpus.

6 Results and Discussion

Table 3 shows in bold the best results achieved for each classifier with each feature selection method. The best accuracy result was 87.15%, which was achieved using PD feature selection with a threshold of 0.125 (which uses 18,149 features or 36% of the total) and FP as a feature weighting method. For comparison, Table 1 shows other results reported in the literature. All approaches used the same dataset which was created by Pang et al. [9] and is the *de facto* standard for sentiment analysis. Note that the evaluation methodology and the number of instances varies between the approaches which makes it difficult to compare the results. Having said that, our best accuracy is 4.55% lower than the best reported result of 91.7% by Abbasi et al.[1].

Our approach offers several key advantages though. Firstly, Abbasi et al's EWGA method is quite computationally expensive. Our best result, though less accurate, is much more computationally efficient, and can make both classification and training faster. Our method is also much simpler and easier to implement. Furthermore we start from a baseline that is 2% lower than Abbasi et al, which reduces the significance of the accuracy difference. The next best accuracy of 91% was achieved by König & Brill [6], who used pattern matching techniques. Their method is also very computationally expensive and has the additional drawback of requiring human intervention. Other approaches in the literature tend to have an accuracy that is similar to ours [7, 5, 9, 8], though without using feature selection.

6.1 Comparison of Classifiers

Figure 1 shows the best accuracy for the two classifiers with all the different feature weighting methods and feature selection methods. For the unigram based feature weights, our results confirm the findings of Pang et al.[9], which is that SVM classifiers are significanly more accurate than NB classifiers. However, for the word group based feature weights the results are less clear. In 8 of the 12 best results for the word group based feature weights, there was less than 0.5% difference between the NB and SVM classifiers, though in the remaining four cases the SVM clearly performed better. This finding shows that while SVM classifiers are substantially more accurate than NB classifiers for unigram based feature weights, they may not necessarily be the best approach for other types of features.

6.2 Comparison of Feature Selectors

Table 3 compares the results between the three feature selectors and the baseline where no feature selection was used for both SVM and NB. The results show that

	PD	SWNSS	SWNPD
0		14,617 (28.71%)	
0.125	18,149 (35.64%)	8,250 (16.2%)	7,433 (14.6%)
0.25	14,860 (29.18%)	7,094 (13.93%)	6,870 (13.49%)
0.375	10,342 (20.31%)	6,061 (11.9%)	5,943 (11.67%)
0.5	9,180 (18.03%)	4,919 (9.66%)	5,750 (11.29%)
0.625	6,716 (13.19%)	3,607 (7.08%)	4,868 (9.56%)
0.75	6,034 (11.85%)	2,302 (4.52%)	4,485 (8.81%)
0.875	5,767 (11.33%)	1,326 (2.6%)	4,431 (8.7%)
1	5,758 (11.31%)	739 (1.45%)	4,431 (8.7%)

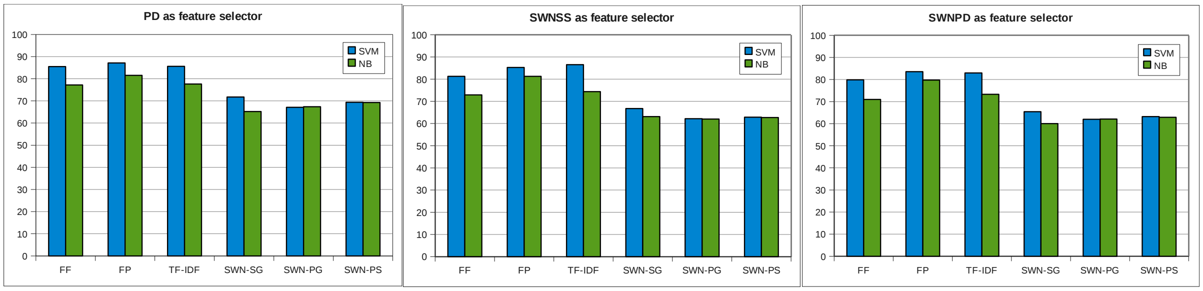
Table 2: Number of selected features by each feature selector for the various selection thresholds.

PD and SWNSS were successful in maintaining classification accuracy when used with appropriate thresholds, and SWNPD was able to maintain accuracy in all cases except for three. PD in particular was able to statistically significantly improve accuracy for nine out of 12 combinations of classifiers and feature weights, while SWNSS and SWNPD were able to improve accuracy in three and one cases respectively. Table 2 shows the number of features selected by each feature selection method at each threshold.

From the results in Table 3 one might conclude that PD was the best feature selection method. However, Figures 2a and 2b provide more information. They show that at low thresholds PD is quite successful at improving accuracy for all of the feature weights, but at higher thresholds accuracy drops sharply. Conversely, both SWNSS and SWNPD have relatively flat lines, indicating that they are more able to find the most effective features at any threshold.

6.3 Comparison of Feature Weights

Figure 2 a), c) and e) show the results for SVM for the three feature selection methods respectively, while Figure 2 b), d) and f) show the same for NB. The x-axis corresponds to the feature selection threshold; as the threshold increases, the number of selected features decreases. The starting point marked with a 'B' corresponds to the baseline where no feature selection is used. In general we found FP was the most accurate feature weighting method, which is in agreement with the results of Pang et al. [9]. Interestingly, the accuracy of FF increased steeply when feature selection was applied. We speculate that this was due to the presence of stop-words, so we conducted a further test of FF with SVM and all words appearing in 1000 or



(a) (b) (c)

Figure 1: Accuracy results (%) for SVM and NB when used with different feature selectors with different thresholds and the six feature weighting methods.

	None	PD	SWNSS	SWNPD
FF	72.5	85.5 ↑ t=0.25	81.3 ↑ t=0.375	79.85 ↑ t=0.125
FP	**85.95**	**87.15** t=0.125	85.3 t=0	**83.55** ↓ t=0.125
TF-IDF	85.9	85.6 t=0.125	**86.55** t=0	82.95 ↓ t=0.125
SWN-SG	65.5	71.75 ↑ t=0.25	66.75 t=0.5	65.45 t=0.125
SWN-PG	62.2	67.1 ↑ t=0.5	62.2 t=0.125	62 t=0.5
SWN-PS	62.85	69.35 ↑ t=0.25	62.85 t=0.375	63.2 t=0.125

(a) SVM Results

	None	PD	SWNSS	SWNPD
FF	68.65	77.2 ↑ t=0.5	72.9 ↑ t=0.875	71 t=0.125
FP	**80.65**	**81.5** t=0.25	**81.3** t=0.125	**79.75** t=0.125
TF-IDF	75.3	77.6 ↑ t=0.25	74.4 t=0.25	73.3 ↓ t=0.125
SWN-SG	59.9	65.2 ↑ t=0.375	63.1 ↑ t=1	60.05 t=0.125
SWN-PG	62	67.35 ↑ t=0.5	62 t=0.125	62.1 t=0.25
SWN-PS	62.7	69.25 ↑ t=0.25	62.7 t=0	62.9 t=0.125

(b) NB Results

Table 3: Comparison between the three feature selection methods and no feature selection for SVM and NB with all six feature weightings. The best accuracy (%) for each feature selector is shown in bold with statistically significant gains over the baseline marked with an up arrow (↑) and statistically significant losses marked with a down arrow (↓).

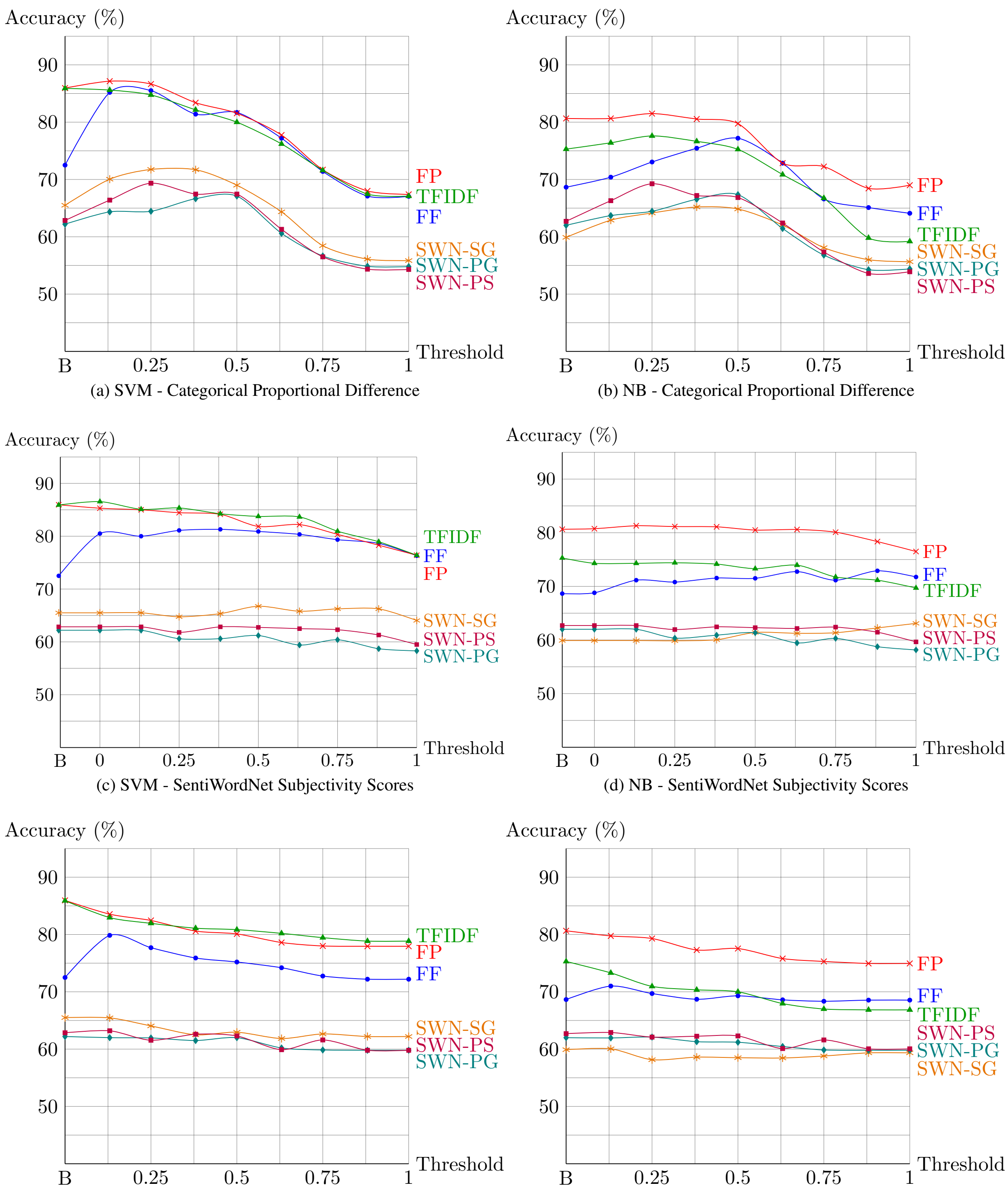
more documents removed. This achieved an accuracy of 83.95%, which indicates that the case for ignoring FF is not as clear cut as the results of Pang et al. [9] suggest.

Unigram based methods consistently outperformed the SWN word group methods for both SVM and NB with all combinations of feature weights and selectors. This finding is in agreement with the findings by Pang et al. [9] and Turney [14], who both noted that summing any out-of-context sentiment scores of individual words does not seem to capture the subtleties that exist in subjective writing. The features produced by SWN-SG, SWN-PG, and SWN-PS illustrate this point quite effectively since they all have approximately equal scores for positive and negative words regardless of the sentiment of the document. This is shown in Figure 3, where we would expect the positive bars to be higher for positive documents and the negative bars to be higher for negative documents. Instead the bars are approximately equal, indicating that there are about as many positive and negative words in positive documents as there are in negative documents.

7 Conclusions

In this paper we empirically and systematically evaluate the performance of a number of feature selection and feature weighting methods for sentiment analysis. In particular, we introduce two new feature selection methods - SWNSS and SWNPD - and compare them, at a number of selection thresholds, with PD, a recently proposed method, shown to be very successful for topic-based classification. We also introduce three feature weighting methods - SWN-SG, SWN-PG and SWN-PS - and compare their performance with the standard and popular FF, FP and TF-IDF methods. The experiments are conducted using two classifiers, SVM and NB, on the movie review data set that has become the *de facto* standard dataset for sentiment analysis.

We achieved an accuracy of 87.15% using PD as a feature selector, FP as a weighting mechanism and SVM as a classifier. This is a promising result as it is comparable with previous state-of-the-art results but is much less computationally expensive. All the feature selectors we tested were able to improve the



(a) SVM - Categorical Proportional Difference

(b) NB - Categorical Proportional Difference

(c) SVM - SentiWordNet Subjectivity Scores

(d) NB - SentiWordNet Subjectivity Scores

(e) SVM - SentiWordNet Proportional Difference

(f) NB - SentiWordNet Proportional Difference

Figure 2: Accuracy results (%) for SVM and NB when used with different feature selectors with different thresholds and the six feature weighting methods.

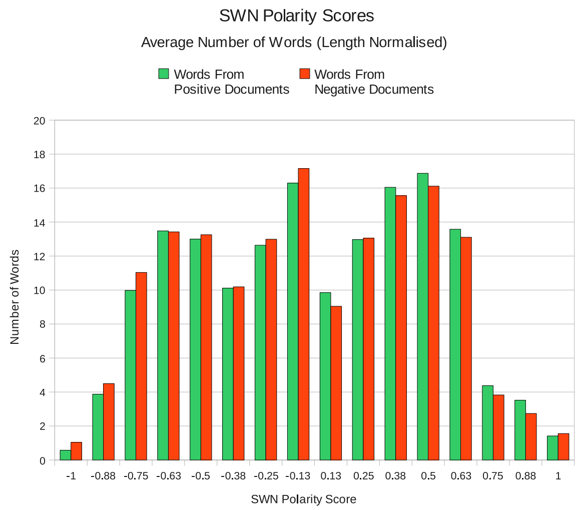


Figure 3: Average number of words with each SWN positive and negative score, from each class of documents.

performance over the baseline without feature selection when used with appropriate weighting methods. Overall, PD was the most successful at improving accuracy, although SWNSS was able to achieve the smallest feature sets whilst maintaining accuracy. The unigram based feature weights - FP, FF and TF-IDF - outperformed SWN-SG, SWN-PG and SWN-PS. Overall, FP was the most successful feature weighting method for both SVM and NB.

Future work will include evaluating more feature selection methods, particularly some of the common ones from text categorisation, such as information gain and χ^2. It would also be valuable to combine some of the feature selectors to see if better feature sets can be produced. Lastly, there would be significant value in repeating these tests on another data set.

References

[1] A Abbasi, HC Chen and A Salem. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions On Information Systems*, Volume 26, Number 3, 2008.

[2] A. Esuli and F. Sebastiani. SentiWordNet: a publicly available lexical resource for opinion mining. In *Proc. of LREC 2006 - 5th Conf. on Language Resources and Evaluation*, Volume 6, 2006.

[3] V. Hatzivassiloglou and K. R. McKeown. Predicting the semantic orientation of adjectives. In *Proc. of the eighth conf. on European chapter of the ACL*, pages 174–181, 1997.

[4] T. Joachims. *Making large-scale support vector machine learning practical, Advances in kernel methods: support vector learning*. MIT Press, Cambridge, MA, 1999.

[5] A. Kennedy and D. Inkpen. Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, Volume 22, Number 2, pages 110–125, May 2006.

[6] A. C König and E. Brill. Reducing the human overhead in text categorization. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 598–603, 2006.

[7] T. Mullen and N. Collier. Sentiment analysis using support vector machines with diverse information sources. In *Proc. of the Conf. on Empirical Methods in Natural Language Processing EMNLP*, pages 412–418, 2004.

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

[9] B. Pang, L. Lee and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *EMNLP '02: Proc. of the ACL-02 conf. on Empirical methods in natural language processing*, pages 79–86. ACL, 2002.

[10] R. Prabowo and M. Thelwall. Sentiment analysis: A combined approach. *Journal of Informetrics*, 2009.

[11] F. Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, Volume 34, Number 1, pages 1–47, 2002.

[12] M. Simeon and R. Hilderman. Categorical proportional difference: A feature selection method for text categorization. In *AusDM*, pages 201–208, 2008.

[13] R. M. Tong. An operational system for detecting and tracking opinions in on-line discussions. In *Working Notes of the ACM SIGIR 2001 Workshop on Operational Text Classification*, pages 1–6, 2001.

[14] P. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *ACL '02: Proc. of the 40th Annual Meeting on ACL*, pages 417–424. ACL, 2002.

[15] I. H Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. San Francisco, Morgan Kaufman Publishers, 2005.