IMPROVING SENTIMENT ANALYSIS WITH PART-OF-SPEECH WEIGHTING

CHRIS NICHOLLS, FEI SONG

Department of Computing and Information Science
University of Guelph, Guelph, Ontario, Canada N1G 2W1
E-MAIL: {cnicholl,fsong}@uoguelph.ca

Abstract:

Sentiment Analysis is concerned with classifying the opinions in a piece of text. We present a term weighting scheme which takes into account Part-of-Speech categories to improve machine learning-based classification of sentiment in product reviews. We experimentally find optimal strengths for each Part-of-Speech category and show that using this weighting method improves overall sentiment classification.

Keywords:

Sentiment Analysis; Feature Selection; Feature Weighting; Part-of-Speech Tagging

1. Introduction

Sentiment Analysis (SA) involves automatically labeling pieces of text according to the sentiment – positive or negative – expressed by the author of the text.

With ever-growing avenues for people to express their views and opinions on the internet, there are many potential applications of SA. Perhaps the most direct would be a system which aggregates and/or summarizes opinions in a collection of text. Such a system could be coupled with a web crawler to collect opinions from blogs and discussion groups, or employed by websites specializing in consumer reviews to improve their existing systems. While consumers may find summaries of reviews useful, it could also be useful to product developers and marketers wanting to follow sentiment towards their goods or services.

Although the primary interest of research is in the review domain, SA has found useful applications in other domains as well. It can be used to extend question answering systems beyond fact-based questions [1]. It can also be used to identify "flames" in online discussions and e-mail [2]. Furthermore, the identification of subjective text, which is closely related to SA, could also be used to aid automatic summarization or information retrieval systems as suggested in [3, 4].

To effectively classify documents by sentiment, the meaning of the subjective terms must be captured. Intuitively, one might expect that adjectives (such as "good", "bad", "poor", "excellent", etc.) would be strong indicators of sentiment. This intuition is, indeed, supported by research in subjective text analysis and SA [3, 4, 5, 6, 7, 8]. Other open Part-of-Speech (POS) categories have also been shown to have some relevance to SA: nouns were used in [8, 9], verbs, in [7, 8], and adverbs were shown to contribute to sentiment and subjectivity in[4, 8, 10].

Our hypothesis is that different POS categories contribute to sentiment in varying degrees. In this work, we use a Maximum Entropy Modeling text classifier to classify the overall sentiment of documents and present a method for weighting words based on POS to improve the classification performance. We then experimentally find optimal weights for each POS category. Although we are focusing on classifying the sentiment of consumer product reviews, we believe that our techniques are applicable to many other domains, such as movie reviews, stock picks, and editorial news articles.

The rest of the paper is organized as follows. Section 2 briefly introduces the motivating and supporting research. Section 3 covers Maximum Entropy Modeling, the machine learning technique we chose for SA. Section 4 describes our new method for SA based on POS weighting. Section 5 explains the experiments we performed, followed by a discussion of our findings in section 6. Finally, we conclude and describe future work in section 7.

2. Related work

Since around 2000, there has been a growing amount of research in SA. Proposed solutions range from linguistic analysis [4, 5], to machine learning techniques [11, 12], to data mining approaches [6, 13, 14]. For brevity, only the approaches which are directly related to the solution we are proposing are detailed in this section.

978-1-4244-3703-0/09/$25.00 ©2009 IEEE

2.1. POS clues

Several methodologies have used terms from a specific POS category for either SA directly, or subjective text classification (which concerns classifying pieces of text as subjective or factual).

Hatzivassiloglou and McKeown [5] developed an unsupervised learning system to learn the semantic orientation (positive or negative) of adjectives. Their system is based on the idea that adjectives connected by conjunctions are likely to have the same orientation, except for adjectives connected by "but" which likely have an opposite orientation. In [3], Hatzivassiloglou and Wiebe developed a method to identify gradable adjectives, that is, adjectives which are intensified or diminished by adverbs. They did not develop a classifier, but show that simply selecting sentences that contain a gradable adjective from their list or an adjective identified in [5] will correctly classify 72% of the subjective sentences in their dataset.

In [9], Riloff et al. presented an unsupervised method using extraction patterns to identify subjective nouns. The learned nouns, along with the adjectives from [4] and [5] were used as features for a Naïve Bayes classifier which classified subjective sentences with 81% precision and 77% recall. Some examples of subjective nouns from our product review dataset used in this paper (described in section 5.1) are "junk", "garbage", "beauty", and "quality".

Benamara et al. [10] made a case for the use of adverbs as clues for SA. They proposed different methods of scoring adverb-adjective combinations and showed that their SA method outperformed some other methods. Some examples of adverbs from our dataset which may convey sentiment are "unfortunately", "poorly", "highly", and "easily".

Chesley et al. [7] used verbs to perform SA on blog posts. They classified verbs into different classes such as *approving, praising, doubting,* or *arguing.* They then used these verb classes, as well as some other features in a Support Vector Machine classifier to classify blog posts as objective, positive or negative. Some examples of subjective verbs from our dataset are "love", "hate", "recommend", and "fail".

Lastly, Yi et al. make use of all of nouns, verbs, adjectives and adverbs in [8]. Their method involves a manually developed lexicon of sentiment expressions which achieved high precision but low recall.

3. Maximum entropy modeling with feature selection

The machine learning text classification technique we chose is Maximum Entropy Modeling (MEM) [15, 16]. There are several reasons why we feel that MEM is a good fit for SA. A key principle of MEM is to agree with everything that is known but carefully avoid assuming anything that is not known [16]. An observation made by Wiebe et al. is that there are a high number of low-frequency words in subjective text [17]. Intuitively, this may be explained as a matter of writing style - people try to avoid using the same descriptive words over and over. Due to the expected low-frequency sentiment carrying words it is likely that many cases will not be covered by our training data and it is important that these cases are weighted equally across different classes.

Furthermore, MEM does not assume feature independence. This in itself can benefit classification tasks, but more importantly, it allows the use of different types of features, and combinations of features. For example, one can mix bag-of-words feature functions, like uni-gram and bi-gram counts, with meta-data features such as the length of a sentence. This versatility can be very useful to classification tasks.

As in text classifiers which classify text according to subject matter, there are many words which will occur across all classes such as "about", "anything", "month", etc. Feature Selection (FS) seeks to reduce the feature space by eliminating features that do not contribute to distinguishing text between different classes. Feature selection not only reduces the computational cost in terms of time and space, but also improves the classification performance. For this work, each word seen in the training set is a feature. Interestingly, a subject matter text classifier usually uses FS to remove subjective words and retain content words, whereas a sentiment analysis classifier needs to use FS to remove content words and retain subjective terms. For our implementation, we apply a supervised FS algorithm called Orthogonal Centroid Feature Selection (OCFS) [18], as it has been shown to work well with MEM [16].

3.1. OCFS

OCFS is aimed at finding a subset of features which maximize the distance between the mean feature vectors (centroids) of all classes. It first computes the centroid vectors for each class, then ranks each feature by its distance from the centroids, and finally, cuts off all features whose distances from the centroids are below a pre-specified threshold. A detailed explanation of OCFS can be found in [18].

3.2. Maximum entropy modeling

The MEM conditional probability distribution takes the following form:

$$p(c|d) = \frac{1}{Z(d)} \exp\left(\sum_i \lambda_i f_i(d,c)\right) \tag{1}$$

where *p(c|d)* is the probability of document *d* having class label *c*, λ_i's are the parameters to be estimated through training using the Improved Iterative Scaling algorithm [15], $f_i($ is a feature function which, in this work, returns a weight for the i^{th} word in the document (defined later in Sections 4.2 and 4.3), and *Z(d)* is a normalizing factor of the form:

$$Z(d) = \sum_c \exp\left(\sum_i \lambda_i f_i(d,c)\right) \tag{2}$$

4. Sentiment analysis with part-of-speech weighting

4.1. Document pre-processing

Several pre-processing steps are performed. First, and most importantly, is POS tagging. In our implementation, we use OpenNLP tools for POS tagging, which is available at http://opennlp.sourceforge.net. Additionally, we apply stemming and stopword removal, but only with the baseline classifier (detailed in Section 5.2.1).

4.2. Word count weighting

To establish our baseline results, we use the weighted feature function in the following form, which is commonly used with MEM for text classification tasks [15, 16]. It is a normalized count of words in a given document:

$$f_{w,c'}(d,c) = \begin{cases} \frac{N(d,w)}{N(d)}, & \text{if } c = c' \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

where *N(d,w)* is the number of times word *w* occurs in document *d* and *N(d)* is the number of words in document *d*.

4.3. Part-of-speech weighting

Initially, the motivation was to have a mechanism to discount content terms and strengthen sentiment carrying terms. The intuition was that nouns are likely to dominate the class of content terms. However, there are many sentiment carrying nouns, as evidenced by [9] – nouns like "quality", "problem", "disappointment" or "value" (from our dataset). As such, arbitrarily cutting out all nouns reduces coverage of the sentiment carrying words to a point where it causes decreased classification performance.

The thought quickly evolved into the idea of weighting words from different POS categories higher (or lower) than others. By doing this, the intention is to get further separation between sentiment carrying words and content words, without completely ignoring entire word categories.

Our proposed weighting scheme is defined as follows. We first manually specify a strength for each POS *str(POS).* For our experiments we use the rules *str(POS)* $\in$ {1,2,3,4,5} to clearly show proportionality between POS category strengths, but any weight could be used. Once this strength has been set, the feature weight is normalized over all POS strengths in the document:

$$f_{w,c'}(d,c) = \begin{cases} \frac{str(POS_w)}{\sum_{i \in d} str(POS_i)}, & \text{if } c = c' \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Note that we filter out all words that are not in an open POS category that we are considering. This means implicitly that if $POS_w \notin$ {noun,verb,adjective,adverb} then $f_{w,c'}(d,c) = 0$.

5. Evaluation

5.1. Dataset

The dataset used for our experiments was originally created by the authors of [6, 13], and is available at http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html. The total set contains product reviews of 14 different products collected from cnet.com and amazon.com.

In its original form, the data set does not have all of the annotations required for our research. Therefore we had to manually review all of the documents and add labels for the overall sentiment of the document as it was believed to be. We also had to remove some bad documents (either incomprehensible or not actual reviews). While these problematic documents could be treated as noise, we did not feel that they were useful for these experiments, especially because we could not manually assign a sentiment label to them. Furthermore, we also had to remove an entire set of reviews for one product, as they were formatted differently from the others and did not have a break between documents.

Our final data set comprised of 574 reviews for 13 different products. All of the products are consumer electronics or digital cameras, except for a few reviews of a power tool (router). Among the total of 574 documents, 399 reviews are positive (69.5%) and 175 are negative

(30.5%).

As will be explained in Section 5.2, the experiments were performed in two passes and required different partitioning of the data.

For the first pass (hereafter referred to as Dataset A) the data was partitioned with the following goals:

- Uniformly distribute positive and negative documents in the training set (50% positive/50% negative)
- Maximize the limited number of negative documents in both training and testing sets
- Maximize the number of documents in the testing set

The resultant data set had 87 negative and 87 positive documents in the training set and 88 negative and 312 positive in the testing set.

The second pass (hereafter referred to as Dataset B) used 10-fold cross-validation. The entire set was broken into 10 folds of 17 randomly selected positive and 17 randomly selected negative documents, resulting in a total of 170 positive and 170 negative in the entire set. The remaining documents were left out.

5.2. Evaluation metrics

The metrics used for evaluation are as follows. Accuracy (the percent of documents which are correctly classified) is calculated for the entire set. Precision (P) and recall (R) are also used and calculated with respect to each of positive and negative classes. For each class, we combine the P and R with the F measure:

$$F = \frac{2PR}{P + R} \qquad (5)$$

The various P, R and accuracy measures are all presented, but the F measure (averaged over the positive and negative classes) is used as the composite metric for measuring the performance.

5.2.1. Baseline Results

For the baseline experiments, our goal was to find the best possible performance of the classifier, using only the OCFS and the word count feature weighting. The set of FS cutoff thresholds used were 500, 1000, 2000, 2500 and all features (no FS). To further tune the classifier, different iteration counts for IIS (Improved Iterative Scaling) were used for each feature cutoff [15]. They were 10, 15, 20, 25, and 30. For the sake of brevity, only the results of best performing number of iterations are presented.

We also compared the use of stopword removal and stemming against a configuration where stemming and stopword removal are not performed, but all closed POS words – everything that is not a noun, verb, adjective or adverb - are filtered out (hereafter referred to as closed POS filter).

Oddly, the baseline classifier yielded identical results with 1000 features and with no FS at all. Table 1 shows the broad-level results (more detail is shown in Table 2).

Table 1. Baseline performance

	Feature cutoff	Accuracy	F (average)
Stemming, stopword	1000 / ALL	79.00%	72.00%
Closed POS filter	500	78.75%	71.18%
Closed POS filter	ALL	76.5%	68.24%

5.2.2. Finding optimal POS strengths

To find optimal POS strengths, the classifier was evaluated using all combinations of $str(POS_i) \in \{1,2,3,4,5\}$ for each of nouns, verbs, adjectives and adverbs. For example, one combination would be: *str(noun)=1, str(verb)=2, str(adjective)=5, str(adverb)=3.* There are 625 possible combinations, and for each combination we use the same set of FS cutoff thresholds and IIS iterations as in the baseline experiment. This experiment uses Dataset A to identify the potential optimal POS strengths as quickly as possible. Table 2 shows the top 10 combinations. The rest of the data in Table 2 is described in the next section.

5.2.3. Cross-validation

To ensure that the classifier was achieving the best possible results, we performed 10-fold cross-validation, using Dataset B, for each of the baseline parameters, using both a 1000 feature cutoff as well as all features. We also performed cross-validation using the 10 best performing POS strength configurations. These configurations as well as the results using cross-validation are shown in Table 2.

Table 2. Optimal POS strengths using 10-fold cross-validation

POS Strength				Feature Cutoff	Accuracy	Positive			Negative			Average
Noun	Verb	Adv.	Adj.			P	R	F	P	R	F	F
Baseline				ALL	75.00%	72.82%	80.59%	76.23%	78.81%	69.41%	73.36%	74.80%
2	3	4	5	1500	79.41%	76.52%	86.47%	80.97%	84.08%	72.35%	77.43%	79.20%
2	3	3	5	1500	79.12%	75.62%	87.06%	80.78%	84.58%	71.18%	77.04%	78.91%
1	2	4	3	1500	78.53%	75.45%	85.88%	80.15%	83.36%	71.18%	76.48%	78.32%
1	3	5	4	ALL	78.24%	76.75%	82.94%	79.36%	81.35%	73.53%	76.65%	78.01%
1	2	3	3	1500	77.94%	74.57%	85.88%	79.62%	83.48%	70.00%	75.82%	77.72%
1	2	3	5	ALL	77.65%	75.77%	82.35%	78.68%	80.72%	72.94%	76.29%	77.49%
1	2	2	5	ALL	77.65%	75.93%	82.35%	78.71%	80.72%	72.94%	76.21%	77.46%
1	2	4	4	1000	77.65%	74.77%	84.71%	79.27%	82.05%	70.59%	75.62%	77.44%
1	2	2	4	ALL	77.65%	75.36%	83.53%	78.92%	81.60%	71.76%	75.88%	77.40%
2	3	5	3	1000	77.35%	73.66%	85.29%	78.87%	83.31%	69.41%	75.41%	77.14%

6. Discussion

The results of the cross-validation experiments show that when using our POS weighting with optimal weights, the performance is increased by almost 5 points over baseline.

The optimal weights, which typically have adjectives and adverbs in the top half of the range of possible strengths, and nouns and verbs in the lower half, are not too surprising. This verifies the intuition that adjectives are very strong indicators of sentiment, but also suggests that adverbs can be just as strong.

After training the classifier with the top performing configuration, we examined the terms which were not cut off by OCFS. In this set of terms, there are obvious cases of adverbs conveying sentiment (“unfortunately”, “nicely”), but there are some subtle cases as well. Words like “too”, for example. When “too” is used to modify an adjective, it is almost always negative in our dataset: “too small”, or “too quiet”. Also, the words “then” and “again” are ranked similarly by OCFS and both have values which indicate a mild negative weight. We believe this suggests that the terms frequently occur together as “then again”, which seems would frequently occur near a negative sentiment.

We can see that out of the 10 top performing POS strength combinations, 6 of them had a feature cutoff of 1000-1500, which suggests that this is a good feature cutoff range for SA. The question that arises, however, is why sometimes no FS is the best FS. A possible explanation for this is that in these cases, there are commonly one or two POS categories that dominate the others in terms of weighting. When this happens the odd low-frequency term that is left in the feature space will contribute little to classification of a document. Therefore, with similar POS strength combinations, it may be beneficial to keep all terms in the feature space.

7. Conclusions and future work

We have presented a method for weighting terms based on POS and shown that it can improve the performance of SA. Furthermore, the optimal POS strengths observed in our experiments can be used to shed some light on the relevance of each POS category to SA. If one looks at the strengths as relevance weighting, then we have shown that adjectives and adverbs are highly relevant to SA (up to 5 times as relevant as nouns), verbs are quite relevant, and nouns are moderately relevant.

While our proposed method helps SA, we believe that it has potential to be even more beneficial to subjectivity classification. Due to the time constraint, we were unable to perform such an evaluation in this paper, but we plan to do so in future research.

FS was shown to be beneficial more often than not, but the techniques we used were developed specifically for subject matter text classification. It may be worthwhile to research other potential FS methods, and perhaps develop one specifically for SA, or possibly try OCFS using the POS weighting scheme presented in this paper.

In this paper, we proposed methods to improve SA in the general sense. Our experiments were set up to evaluate our classification method. However, a real world SA system has to do much more than classify sentiment at the document level. A fully automated system would need to identify all of the topics in a document, group subjective information about each topic, classify that sentiment and further relate the topics to topics found in other documents. All of those tasks provide many future research avenues.

Acknowledgements

The authors would like to acknowledge the financial support of CITO, a division of Ontario Centres of Excellence (OCE) in Canada.

References

[1] C. Cardie, J. Wiebe, T. Wilson and D. Litman, “Combining low-level and summary representations of opinions for multi-perspective question answering”, in *Working Notes-New Directions in Question Answering (AAAI Spring Symposium Series),* 2003.

[2] E. Spertus, “Smokey: Automatic recognition of hostile messages”, in *Proceedings of the National Conference on Artificial Intelligence*, pp. 1058-1065, 1997.

[3] J. M. Wiebe, “Learning subjective adjectives from corpora”, in *Proceedings of the National Conference on Artificial Intelligence,* 2000, pp. 735-741.

[4] V. Hatzivassiloglou and J. M. Wiebe, “Effects of adjective orientation and gradability on sentence subjectivity”, in *Proceedings of the 18th Conference on Computational Linguistics-Volume 1*, pp. 299-305, 2000.

[5] V. Hatzivassiloglou and K. R. McKeown, “Predicting the semantic orientation of adjectives”, in *Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics*, pp. 174-181, 1997.

[6] M. Hu and B. Liu, “Mining opinion features in customer reviews”, in *Proceedings of the National Conference on Artificial Intelligence*, pp. 755-760, 2004.

[7] P. Chesley, B. Vincent, L. Xu and R. K. Srihari, “Using verbs and adjectives to automatically classify blog sentiment”, *AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW 2006),* vol. 580, pp. 233, 2006.

[8] J. Yi, T. Nasukawa, R. Bunescu, W. Niblack, I. B. M. A. R. Center and C. San Jose, “Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques”, in *Third IEEE International Conference on Data Mining (ICDM 2003)*, pp. 427-434, 2003.

[9] E. Riloff, J. Wiebe and T. Wilson, “Learning subjective nouns using extraction pattern bootstrapping”, in *Proceedings of the 7th Conference on Natural Language Learning (CoNLL-2003)* , pp. 25–32, 2003.

[10] F. Benamara, C. Cesarano, A. Picariello, D. Reforgiato and V. Subrahmanian, “Sentiment analysis: Adjectives and adverbs are better than adjectives alone”, in *Proceedings of the International Conference on Weblogs and Social Media (ICWSM),* 2007.

[11] B. Pang, L. Lee and S. Vaithyanathan, “Thumbs up?: Sentiment classification using machine learning techniques”, in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing-Volume 10*, pp. 79-86, 2002.

[12] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. A. Reis and J. Reynar, “Building a sentiment summarizer for local service reviews”, in *WWW Workshop on NLP in the Information Explosion Era,* 2008.

[13] M. Hu and B. Liu, “Mining and summarizing customer reviews”, in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 168-177, 2004.

[14] S. Morinaga, K. Yamanishi, K. Tateishi and T. Fukushima, “Mining product reputations on the web”, in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 341-349, 2002.

[15] K. Nigam, J. Lafferty and A. McCallum, “Using maximum entropy for text classification”, in *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*, pp. 61-67, 1999.

[16] J. Cai and F. Song, “Maximum Entropy Modeling with Feature Selection for Text Categorization”, *Lecture Notes in Computer Science,* vol. 4993, pp. 549-554, 2008.

[17] J. Wiebe, T. Wilson and M. Bell, “Identifying collocations for recognizing opinions”, in *Proceedings of the ACL-01 Workshop on Collocation: Computational Extraction, Analysis, and Exploitation*, pp. 24–31, 2001.

[18] J. Yan, N. Liu, B. Zhang, S. Yan, Z. Chen, Q. Cheng, W. Fan and W. Y. Ma, “OCFS: Optimal orthogonal centroid feature selection for text categorization”, in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 122-129, 2005.