USNA: A Dual-Classifier Approach to Contextual Sentiment Analysis

Ganesh Harihara and **Eugene Yang** and **Nathanael Chambers**
United States Naval Academy
Annapolis, MD 21401, USA
nchamber@usna.edu

Abstract

This paper describes a dual-classifier approach to contextual sentiment analysis at the SemEval-2013 Task 2. Contextual analysis of polarity focuses on a word or phrase, rather than the broader task of identifying the sentiment of an entire text. The Task 2 definition includes target word spans that range in size from a single word to entire sentences. However, the context of a single word is dependent on the word's surrounding syntax, while a phrase contains most of the polarity within itself. We thus describe separate treatment with two independent classifiers, outperforming the accuracy of a single classifier. Our system ranked 6th out of 19 teams on SMS message classification, and 8th of 23 on twitter data. We also show a surprising result that a very small amount of word context is needed for high-performance polarity extraction.

1 Introduction

A variety of approaches to sentiment analysis have been proposed in the literature. Early work sought to identify the general sentiment of entire documents, but a recent shift to social media has provided a large quantity of publicly available data, and private organizations are increasingly interested in how a population "feels" toward its products. Identifying the polarity of language toward a particular topic, however, no longer requires identifying the sentiment of an entire text, but rather the *contextual sentiment* surrounding a target phrase.

Identifying the polarity of text toward a phrase is significantly different from a sentence's overall polarity, as seen in this example from the SemEval-2013 Task 2 (Wilson et al., 2013) training set:

> *I had a severe* ***nosebleed*** *last night.* ***I think my iPad caused it as I was browsing for a few hours on it.*** *Anyhow, its stopped, which is good.*

An ideal sentiment classifier would classify this text as overall positive (the nosebleed stopped!), but this short snippet actually contains three types of polarity (positive, negative, and neutral). The middle sentence about the iPad is not positive, but neutral. The word 'nosebleed' has a very negative polarity in this context, and the phrase 'its stopped' is positive. Someone interested in specific health concerns, such as nosebleeds, needs a contextual classifier to identify the desired polarity in this context.

This example also illustrates how phrases of different sizes require unique handling. Single token phrases, such as 'nosebleed', are highly dependent on the surrounding context for its polarity. However, the polarity of the middle iPad sentence is contained within the phrase itself. The surrounding context is not as important. This paper thus proposes a dual-classifier that trains two separate classifiers, one for single words, and another for phrases. We empirically show that unique features apply to both, and both benefit from independent training. In fact, we show a surprising result that a very small window size is needed for the context of single word phrases. Our system performs well on the SemEval task, placing 8th of 23 systems on twitter text. It also shows strong generalization to SMS text messages, placing 6th of 19.

*Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 390–394, Atlanta, Georgia, June 14-15, 2013.

2 Previous Work

Sentiment analysis is a large field applicable to many genres. This paper focuses on social media (microblogs) and contextual polarity, so we only address the closest work in those areas. For a broader perspective, several survey papers are available (Pang and Lee, 2008; Tang et al., 2009; Liu and Zhang, 2012; Tsytsarau and Palpanas, 2012).

Microblogs serve as a quick way to measure a large population's mood and opinion. Many different sources have been used. O'Connor et al. (2010) used Twitter data to compute a ratio of positive and negative words to measure consumer confidence and presidential approval. Kramer (2010) counted lexicon words on Facebook for a general 'happiness' measure, and Thelwall (2011) built a general sentiment model on MySpace user comments. These are general sentiment algorithms.

Specific work on microblogs has focused on finding noisy training data with distant supervision. Many of these algorithms use emoticons as semantic indicators of polarity. For instance, a tweet that contains a sad face likely contains a negative polarity (Read, 2005; Go et al., 2009; Bifet and Frank, 2010; Pak and Paroubek, 2010; Davidov et al., 2010; Kouloumpis et al., 2011). In a similar vein, hashtags can also serve as noisy labels (Davidov et al., 2010; Kouloumpis et al., 2011). Most work on *distant supervision* relies on a variety of syntactic and word-based features (Marchetti-Bowick and Chambers, 2012). We adopt many of these features.

Supervised learning for *contextual* sentiment analysis has not been thoroughly investigated. Labeled data for specific words or queries is expensive to generate, so Jiang et al. (2011) is one of the few approaches with labeled training data. Earlier work on product reviews sought the sentiment toward particular product features. These systems used rule based approaches based on parts of speech and other surface features (Nasukawa and Yi, 2003; Hu and Liu, 2004; Ding and Liu, 2007).

Finally, topic identification in microblogs is also related. The first approaches are somewhat simple, selecting single keywords (e.g., "Obama") to represent the topic (e.g., "US President"), and retrieve tweets that contain the word (O'Connor et al., 2010; Tumasjan et al., 2010; Tan et al., 2011). These systems then classify the polarity of *the entire tweet*, and ignore the question of polarity toward the particular topic. This paper focuses on the particular keyword or phrase, and identifies the sentiment toward that phrase, not the overall sentiment of the text.

3 Dataset

This paper uses three polarity classes: positive, negative, and neutral. We developed all algorithms on the 'Task A' corpora provided by SemEval-2013 Task 2 (Wilson et al., 2013). Both training and development sets were provided, and an unseen test set was ultimately used to evaluate the final systems. The number of tweets in each set are shown here:

	positive	negative	neutral
training	5348	2817	422
development	648	430	57
test (tweet)	2734	1541	160
test (sms)	1071	1104	159

4 Contextual Sentiment Analysis

Contextual sentiment analysis focuses on the disposition of a certain word or groups of words. Most data-driven approaches rely on a labeled corpus to drive the learning process, and this paper is no different. However, we propose a novel approach to contextual analysis that differentiates between *single words* and *phrases*.

The semantics of a single word in context from that of a phrase are fundamentally different. Since one word will have multiple contexts and is heavily influenced by the surrounding words, more consideration is given to adjacent words. A phrase often carries its own semantics, so has less variability in its meaning based on its context. Context is still important, but we propose separate classifiers in order to learn weights unique to tokens and phrases. The following describes the two unique feature sets. We trained a Maximum Entropy classifier for each set.

4.1 Text Pre-Processing

All text is lowercased, and twitter usernames (e.g., @*user*) and URLs are replaced with placeholder tokens. The text is then split on whitespace. We also prepend the occurrence of token "not" to the subsequent token, merging the two (e.g., "not happy" be-

comes "not-happy"). We also found that removing prefix and affix punctuation from each token, and storing the punctuation for later use in punctuation features boosts performance. These cleaned tokens are the input to the features described below.

4.2 Single Word Sentiment Analysis

Assigning polarity to a single word mainly requires features that accurately capture the surrounding context. In fact, many single words do no carry any polarity in isolation, but solely require context. Take the following two examples:

> ***Justin*** *LOVE YA so excited for the concert in october MEXICO LOVES YOU*
>
> *Im not getting on twitter tomorrow because all my TL will consist of is a bunch of girls talking about* ***Justin*** *Beiber*

In these examples, Justin is the name of a singer who does not carry an initial polarity. The first tweet is clearly positive toward him, while the second is not. Our single-token classifier used the following set of features to capture these different contexts:

Target Token: The first features are the unigram and bigram ending with the target token. We attach a unique string to each to distinguish it from the text's other n-grams. We also include a feature for any punctuation that was attached to the end of the token (e.g., 'Justin!' generates '!' as a feature).

Target Patterns: This feature generalizes the n-grams that include the target word. It replaces the target word with a variable in an effort to capture general patterns that indicate sentiment. For instance, using the first tweet above, we add the trigram '<*s*> _ *LOVE*' and two bigrams, '<*s*> _' and '_ *LOVE*'.

Unigrams, Bigrams, Trigrams: We include all other n-grams in the text within a window of size n from the target token.

Dictionary Matching: We have two binary features, *postivemood* and *negativemood*, that indicate if any word in the text appears in a sentiment lexicon's positive or negative list. We use Bing Liu's Opinion Lexicon[1].

[1] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html\#lexicon

Punctuation Features: We included a binary feature for the presence or absence of exclamation marks anywhere in the text. Further, we generate a feature for punctuation at the end of the text.

Emoticons: We included two binary features for the presence or absence of a smiley face and sad face emoticon.

4.3 Phrasal Sentiment Analysis

We adopted several single word features for use in phrases, including punctuation, dictionary matching, and emoticons. However, since phrasal analysis is often less dependent on context and more dependent on the phrase itself, we altered the n-gram features to be unique to the phrase. The following features are solely used for target phrases, not single words:

Unigrams, Bigrams, Trigrams: We include all n-grams *in the target phrase* only. This differs from the single token features that included n-grams from a surrounding window.

Phrasal Punctuation: If the *target phrase* ends with any type of punctuation, we include it as a feature.

5 Experiments

Initial model design and feature tuning was conducted on the SemEval-2013 Task 2 training set for training, and its dev set for evaluation. We split the data into two parts: tweets with single word targets, and tweets with target phrases. We trained two MaxEnt classifiers using the Stanford JavaNLP toolkit[2]. Each datum in the test set is labeled using the appropriate classifier based on the target phrase's length.

The first experiments are ablation over the features described in Section 4, separately improving the single token and phrasal classifiers. Results are reported in Table 1 using simple accuracy on the development set. We initially do not split off punctuation, and use only unigram features for phrases. The window size is initally infinite (i.e., the entire text is used for n-grams). Bigrams and trigrams hurt performance and are not shown. Reducing the window size to a single token (ignore the entire tweet) increased performance by 1.2%, and stripping punctuation off tokens by another 1.9%. The perfor-

[2] http://nlp.stanford.edu/software/index.shtml

Single Token Features

Just Unigrams	70.5
+ Target Token Patterns	70.4
+ Sentiment Lexicon	71.5
+ Target Token N-Grams	73.3
+ EOS punctuation	73.2
+ Emoticons	73.3
Set Window Size = 1	74.5
Strip punctuation off tokens	76.4

Phrasal Features

Just Unigrams	76.4
+ Emoticons	76.3
+ EOS punctuation	76.6
+ Exclamation Marks	76.5
+ Sentiment Lexicon	77.7

Table 1: Feature ablation in order. Single token features begin with unigrams only, holding phrasal features constant at unigrams only. The phrasal table picks up where the single token table finishes. Each row uses all features added in previous rows.

Dual-Classifier Comparison

Single Classifier	76.6%
Dual-Classifier	77.7%

Table 2: Performance increase from splitting into two classifiers. Accuracy reported on the development set.

mance increase with phrasal features is 1.3% absolute, whereas token features contributed 5.9%.

After choosing the optimum set of features based on ablation, we then retrained the classifiers on both the training and development sets as one large training corpus. The SemEval-2013 Task 2 competition included two datasets for testing: tweets and SMS messages. Official results for both are given in Table 3 using the F1 measure.

Finally, we compare our dual-classifier to a single standard classifier. We use the same features used in Table 1, train on the training set, and report accuracy on the development set. See Table 2. Our dual classifier improves relative accuracy by 1.4%.

6 Discussion

One of the main surprises from our experiments was that a large portion of text could be ignored without hurting classification performance. We reduced

Twitter Dataset

	F1 Score
Top System (1st)	88.9
This Paper (8th)	**81.3**
Majority Baseline (20th)	61.6
Bottom System (24th)	34.7

SMS Dataset

	F1 Score
Top System (1st)	88.4
This Paper (6th)	**79.8**
Majority Baseline (19th)	47.3
Min System (20th)	36.4

Table 3: Performance on Twitter and SMS Data.

the window size in which n-grams are extracted to size one, and performance actually increases 1.2%. At least for single word target phrases, including n-grams of the entire tweet/sms is not helpful. We only used n-gram patterns that included the token and its two immediate neighbors. A nice side benefit is that the classifier contains fewer features, and trains faster as a result.

The decision to use two separate classifiers helped performance, improving by 1.4% relative accuracy on the development set. The decision was motivated by the observation that the polarity of a token is dependent on its surrounding context, but a longer phrase is dependent more on its internal syntax. This allowed us to make finer-grained feature decisions, and the feature ablation experiments suggest our observation to be true. Better feature weights are ultimately learned for the unique tasks.

Finally, the feature ablation experiments revealed a few key takeaways for feature engineering: bigrams and trigrams hurt classification, using a window size is better than the entire text, and punctuation should always be split off tokens. Further, a sentiment lexicon reliably improves both token and phrasal classification.

Opportunities for future work on contextual analysis exist in further analysis of the feature window size. Why doesn't more context help token classification? Do n-grams simply lack the deeper semantics needed, or are these supervised algorithms still suffering from sparse training data? Better sentence and phrase detection may be a fruitful focus.

References

Albert Bifet and Eibe Frank. 2010. Sentiment knowledge discovery in twitter streaming data. In *Lecture Notes in Computer Science*, volume 6332, pages 1–15.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*.

Xiaowen Ding and Bing Liu. 2007. The utility of linguistic rules in opinion mining. In *Proceedings of SIGIR-2007*, pages 23–27.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. Technical report.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the Association for Computational Linguistics (ACL-2011)*.

Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore. 2011. Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*.

Adam D. I. Kramer. 2010. An unobtrusive behavioral model of 'gross national happiness'. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI 2010)*.

Bing Liu and Lei Zhang. 2012. A survey of opinion mining and sentiment analysis. *Mining Text Data*, pages 415–463.

Micol Marchetti-Bowick and Nathanael Chambers. 2012. Learning for microblogs with distant supervision: Political forecasting with twitter. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*.

Tetsuya Nasukawa and Jeonghee Yi. 2003. Sentiment analysis: capturing favorability using natural language processing. In *Proceedings of K-CAP*.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. 2010. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the AAAI Conference on Weblogs and Social Media*.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference On Language Resources and Evaluation (LREC)*.

B. Pang and L. Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*.

Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop (ACL-2005)*.

Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. 2011. User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

H. Tang, S. Tan, and X. Cheng. 2009. A survey on sentiment detection of reviews. *Expert Systems with Applications*.

Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. 2011. Sentiment in twitter events. *Journal of the American Society for Information Science and Technology*, 62(2):406–418.

M. Tsytsarau and T. Palpanas. 2012. Survey on mining subjective data on the web. *Data Mining and Knowledge Discovery Journal*, 24(3):478–514.

Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, and Isabell M. Welpe. 2010. Election forecasts with twitter: How 140 characters reflect the political landscape. *Social Science Computer Review*.

Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Alan Ritter, Sara Rosenthal, and Veselin Stoyanov. 2013. Semeval-2013 task 2: Sentiment analysis in twitter. In *Proceedings of the 7th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.