# SAP-RI: Twitter Sentiment Analysis in Two Days

**Akriti Vij**[1,2]*, **Nishtha Malhotra**[1,2]*, **Naveen Nandan**[1] and **Daniel Dahlmeier**[1]
[1]SAP Research & Innovation, Singapore
[2]Nanyang Technological University, Singapore
{akriti.vij,nishtha.malhotra,naveen.nandan,d.dahlmeier}@sap.com

## Abstract

We describe the submission of the SAP Research & Innovation team to the SemEval 2014 Task 9: Sentiment Analysis in Twitter. We challenged ourselves to develop a competitive sentiment analysis system within a very limited time frame. Our submission was developed in less than two days and achieved an $F_1$ score of 77.26% for contextual polarity disambiguation and 55.47% for message polarity classification, which shows that rapid prototyping of sentiment analysis systems with reasonable accuracy is possible.

## 1 Introduction

Microblogging platforms and social networks have become increasingly popular for expressing opinions on a wide range of topics, hence making them valuable and ever-growing logs of public sentiment. This has motivated the development of automatic natural language processing (NLP) methods to analyse the sentiment expressed in these short, informal messages (Liu, 2012; Pang and Lee, 2008).

In particular, the study of sentiment and opinions in messages from the Twitter microblogging platform has attracted a lot of interest (Jansen et al., 2009; Pak and Paroubek, 2010; Barbosa and Feng, 2010; O'Connor et al., 2010; Bifet et al., 2011). However, comparative evaluations of sentiment analysis of Twitter messages have previously been hindered by the lack of a large benchmark data set. The goal of the SemEval 2013 task 2: Sentiment Analysis in Twitter (Nakov et al., 2013) and this year's continuation in the SemEval 2014 task 9: Sentiment Analysis in Twitter (Rosenthal et al., 2014) is to close this gap by hosting a shared task competition which provided a large corpus of Twitter messages which are annotated with sentiment polarity labels. The task consists of two subtasks: in subtask A *contextual polarity disambiguation*, participants need to predict the polarity of a given word or phrase in the context of a tweet message, in subtask B *message polarity classification*, participants need to predict the dominating sentiment of the complete message. Both tasks consider sentiment analysis to be a three-way classification problem between positive, negative, and neutral sentiment.

In this paper, we describe the submission of the SAP-RI team to the SemEval 2014 task 9. We challenged ourselves to develop a competitive sentiment analysis system within a very limited time frame. The complete system was implemented within only two days. Our system is based on supervised classification with support vector machines with lexical and dictionary-based features. Our system achieved an $F_1$ score of 77.26% for contextual polarity disambiguation and 55.47% for message polarity classification. Although our scores are about 10-20% behind the top-scoring systems, we show that it is possible to develop sentiment analysis systems via rapid prototyping with reasonable accuracy in a very short amount of time.

## 2 Methods

Our system is based on supervised classification with support vector machines and a variety of lexical and dictionary-based features. From the beginning, we decided to restrict ourselves to supervised classification and to focus on the constrained system setting. Experiments with more data or semi-supervised learning would have required additional time and the results of last year's task

* The work was done during an internship at SAP.





*Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 522–526,
Dublin, Ireland, August 23-24, 2014.

did not show any convincing improvements using from additional unconstrained data (Nakov et al., 2013). We cast sentiment analysis as a multi-class classification problem with three classes: *positive*, *negative*, and *neutral*. For the features, we tried to re-implement most of the features from the NRC-Canada system (Mohammad et al., 2013) which was the best performing system in last year's task. We describe the features in the following sections.

### 2.1 Task A : Features

For the contextual polarity disambiguation task, we extract features from the target phrase itself and from a surrounding word window of four words before and after the target phrase. To handle negation, we append the suffix *-neg* to all words in a negated context. A negated context includes any word in the target phrase or context that is following a negation word [1] up to the next following punctuation symbol.

- **Word N-grams:** all lowercased unigrams and bigrams from the target phrase and the context. We extract the lowercased full string of the target phrase as an additional feature.
- **Character N-grams:** lowercased character bigram and trigram prefixes and suffixes from all words in the target phrase and the context.
- **Elongations**: binary feature that indicates the presence of one or more words in the target phrase or context that have a letter repeated for 3 for more times e.g., *coool*.
- **Emoticons:** two binary features that indicate the presence of positive or negative emoticons in the target phrase or the context, respectively. Two additional binary features indicate the presence of positive or negative emoticons at the end of the target phrase or context[2].
- **Punctuation:** three count features for the number of tokens that consist only of exclamation marks, only of questions marks, or a mix of exclamation marks and questions marks, in the target phrase and context, receptively.
- **Casing:** two binary features that indicate the presence of at least one all upper-case word and at least one title-cased word in the target phrase or context, respectively.
- **Stop words:** a binary feature that indicates if all the words in the target phrase or context are stop words. If so, an additional feature indicates the number of stop words: 1, 2, 3, or more stop words.
- **Length:** the number of tokens in the target phrase and the context, plus a binary feature that indicates the presence of any word with more than three characters.
- **Position:** three binary features that indicate whether a target phrase is at the beginning, in the middle, or at the end of the tweet.
- **Hashtags:** all hashtags in the target phrase or the context. To handle hashtags which are formed by concatenating words, e.g., *#ihatemondays*, we additionally split hashtags using a simple dictionary-based approach and add each token of the segmented hashtag as an additional features.
- **Twitter user:** binary feature that indicates whether the context or the target phrase contain a mention of a Twitter user.
- **URL:** binary feature that indicates whether the context or the target phrase contains a URL.
- **Brown cluster:** the word cluster index for each word in the context. Cluster indexes are obtained from the Brown word clusters of the ARK Twitter tagger (Owoputi et al., 2013).
- **Sentiment lexicons:** we add the following sentiment dictionary features for the target phrase and the context for four different sentiment lexicons (NRC sentiment lexicon, NRC Hashtag lexicon (Mohammad et al., 2013), MPQA sentiment lexicon (Wilson et al., 2005), and Bing Liu lexicon (Hu and Liu, 2004)):
  - the count of words with positive sentiment score.
  - the sum of the sentiment scores for all words.

[1] `http://sentiment.christopherpotts.net/lingstruc.html`

[2] positive emoticons: *:-), :), :B, :-B, :3, =), <3, :D, :-D, =D, :'), :d, ;), :}, :], :P, :-P, :-p, :p*. negative emoticons: *:-(, :/, :{, :[, -.-, -_-, :O, :o, :(, :x, :X, v.v, ;(*

– the maximum non-negative sentiment score for any word.
– the sentiment score of the last word with positive sentiment score.

We extract these features for both the target phrase and the context. For words that are marked as negated, the sign of the sentiment scores flipped. The MPQA lexicons requires part of speech information. We use the ARK Twitter part-of-speech tagger (Owoputi et al., 2013) to tag the input with part of speech tags.

### 2.2 Task B : Features

For the message polarity task, we extract features from the entire tweet message. The features are similar to the features for phrase polarity disambiguation. As before we handle negation by appending the suffix *-neg* to all words that appear in a negated context.

- **Word N-grams:** all lowercased N-grams for N=1, . . . , 4 from the message. We also include "skipgrams" for each N-gram by replacing each token in the N-gram with a asterisk place holder, e.g., *the_cat* → **_cat, the_**.
- **Character N-grams:** lowercased character level N-grams for N=3, . . . , 5 for all the words in the message. Character N-grams do not cross word boundaries.
- **Elongations**: count of words in the message which have a letter repeated for 3 for more times.
- **Emoticons:** similar to the contextual polarity disambiguation task: two binary features for presence of positive or negative emoticons in the message and two binary features indicate the presence of positive or negative emoticons at the end of the message.
- **Punctuation:** similar to the contextual polarity disambiguation task: three count features for the number of tokens that consist only of exclamation marks, only of questions marks, or a mix of exclamation marks and questions marks.
- **Hashtags:** all hashtags in the message. We do not split concatonated hashtags.

| | # Tokens | # Tweets |
|---|---|---|
| **Subtask A** | | |
| Training (SemEval 2014 train) | 160,992 | 7,884 |
| Development (SemEval 2013 test) | 76,409 | 3,710 |
| **Subtask B** | | |
| Training (SemEval 2014 train) | 139,128 | 7,112 |
| Development (SemEval 2013 test) | 47,052 | 2,405 |

Table 1: Overview of the data sets.

- **Casing:** the count of all upper-case words in the message.
- **Brown cluster:** similar to the contextual polarity disambiguation task: the cluster index for each word in the message.

## 3 Experiment and Results

In this section, we report experimental result for our method. We used the scikit-learn Python machine learning library (Pedregosa et al., 2011) to implement the feature extraction pipeline and the support vector machine classifier. We use a linear kernel for the support vector machine and fixed the SVM hyper-parameter $C$ to 1.0. We found that scikit-learn allowed us to implement the system faster and resulted in much more compact code than other machine learning tools we have worked with in the past.

We used the official training set provided for the SemEval 2014 task to train our system and evaluated on the test set of the SemEval 2013 task which served as development data for this year's task [3]. Tweets in the training data that were not available any more through the Twitter API were removed from the training set. An overview of the data sets is shown in Table 1. For the evaluation, we compute precision, recall and $F_1$ measure for the positive, negative, and neutral sentiment tweets. Following the official evaluation metric, the overall precision, recall, and $F_1$ measure of the system is the average of the precision, recall, and $F_1$ measures for positive and negative sentiment, respectively.

Here, we report a feature ablation study: we omitted each individual feature category from the complete feature set to determine its influence on the overall performance. Table 2 summarizes the results for subtask A and B. Surprisingly many of the features do not result in a reduction of the $F_1$ score when removed, or even increase the score,

[3] We also did some experiments with a 60:40 training/test split of the SemEval 2014 training data which showed comparable results

| Features | Positive | | | Negative | | | Neutral | | | Overall | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Subtask A** | P | R | $F_1$ | P | R | $F_1$ | P | R | $F_1$ | P | R | $F_1$ |
| All features | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.82 | 0.83 | 0.82 |
| w/o Word N-grams | 0.84 | 0.82 | 0.83 | 0.71 | 0.74 | 0.72 | 0.14 | 0.16 | 0.15 | 0.77 | 0.78 | 0.78 |
| w/o character N-grams | 0.85 | 0.89 | 0.87 | 0.80 | 0.78 | 0.79 | 0.27 | 0.12 | 0.17 | 0.82 | 0.83 | 0.83 |
| w/o elongation | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.81 | 0.82 | 0.81 |
| w/o emoticons | 0.85 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.24 | 0.14 | 0.18 | 0.82 | 0.83 | 0.82 |
| w/o punctuation | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.81 | 0.83 | 0.82 |
| w/o casing | 0.86 | 0.87 | 0.87 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.82 | 0.83 | 0.82 |
| w/o stop words | 0.86 | 0.87 | 0.86 | 0.78 | 0.79 | 0.78 | 0.24 | 0.15 | 0.18 | 0.82 | 0.83 | 0.82 |
| w/o length | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.14 | 0.17 | 0.82 | 0.83 | 0.82 |
| w/o position | 0.86 | 0.87 | 0.86 | 0.77 | 0.78 | 0.78 | 0.24 | 0.13 | 0.17 | 0.81 | 0.83 | 0.82 |
| w/o hashtags | 0.86 | 0.87 | 0.87 | 0.78 | 0.78 | 0.78 | 0.24 | 0.14 | 0.18 | 0.82 | 0.83 | 0.82 |
| w/o twitter user | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.82 | 0.83 | 0.82 |
| w/o URL | 0.86 | 0.87 | 0.86 | 0.78 | 0.78 | 0.78 | 0.23 | 0.13 | 0.17 | 0.81 | 0.82 | 0.81 |
| w/o Brown cluster | 0.86 | 0.88 | 0.87 | 0.78 | 0.80 | 0.79 | 0.25 | 0.13 | 0.17 | 0.82 | 0.84 | 0.83 |
| w/o Sentiment lexicon | 0.81 | 0.84 | 0.82 | 0.70 | 0.68 | 0.69 | 0.16 | 0.09 | 0.11 | 0.75 | 0.76 | 0.76 |
| **Subtask B** | | | | | | | | | | | | |
| All features | 0.81 | 0.54 | 0.65 | 0.66 | 0.34 | 0.44 | 0.59 | 0.89 | 0.71 | 0.74 | 0.44 | 0.54 |
| w/o word N-grams | 0.73 | 0.59 | 0.65 | 0.52 | 0.46 | 0.49 | 0.61 | 0.75 | 0.67 | 0.62 | 0.52 | 0.57 |
| w/o character N-grams | 0.80 | 0.49 | 0.61 | 0.65 | 0.23 | 0.34 | 0.56 | 0.90 | 0.69 | 0.72 | 0.36 | 0.48 |
| w/o elongation | 0.81 | 0.54 | 0.65 | 0.66 | 0.34 | 0.44 | 0.59 | 0.89 | 0.71 | 0.74 | 0.44 | 0.55 |
| w/o emoticons | 0.82 | 0.54 | 0.65 | 0.66 | 0.33 | 0.44 | 0.59 | 0.89 | 0.72 | 0.74 | 0.44 | 0.55 |
| w/o punctuation | 0.81 | 0.54 | 0.65 | 0.66 | 0.34 | 0.45 | 0.59 | 0.89 | 0.71 | 0.74 | 0.44 | 0.55 |
| w/o casing | 0.81 | 0.54 | 0.65 | 0.66 | 0.33 | 0.44 | 0.59 | 0.89 | 0.71 | 0.74 | 0.44 | 0.55 |
| w/o hashtags | 0.82 | 0.54 | 0.65 | 0.65 | 0.33 | 0.44 | 0.59 | 0.89 | 0.71 | 0.74 | 0.44 | 0.54 |
| w/o Brown cluster | 0.81 | 0.54 | 0.65 | 0.65 | 0.33 | 0.44 | 0.59 | 0.89 | 0.71 | 0.73 | 0.43 | 0.54 |

Table 2: Experimental Results for feature ablation study. Each row shows the precision, recall, and $F_1$ score for the positive, negative, and neutral class and the overall precision, recall, and $F_1$ score after removing the particular feature from the features set.

although not significantly. The most effective features are word N-grams and the sentiment lexicons. It is interesting that the performance for the neutral class is very low for subtask A and high for subtask B. We can also see that for subtask B, our system clearly has a problem with recall for the positive and negative sentiment.

For the performance of our system in the SemEval 2014 shared task, we report the official overall $F_1$ scores of our system as released by the organizers on the official test set in Table 3. The scores were reported separately for different test sets: the SemEval 2013 Twitter test set, a new SemEval 2014 Twitter test set, a new test set from LiveJournal blogs, the SMS test set from the NUS SMS corpus (Chen and Kan, 2012), and a new test set of sarcastic tweets. We also include the $F_1$ score of the best participating system for each test set and the rank of our system among all participating systems. The results of our system were fairly robust across different domains, with the exception of messages containing sarcasm which shows understanding sarcasm requires a deeper and more subtle understanding of the text that is not captured well in a simple linear model.

| Dataset | Best score | Our score | Rank |
|---|---|---|---|
| **Subtask A** | | | |
| LiveJournal 2014 | 85.61 | 77.68 | 18 / 27 |
| SMS 2013 | 89.31 | 80.26 | 13 / 27 |
| Twitter 2013 | 90.14 | 80.32 | 17 / 27 |
| Twitter 2014 | 86.63 | 77.26 | 15 / 27 |
| Twitter 2014 Sarcasm | 82.75 | 70.64 | 14 / 27 |
| **Subtask B** | | | |
| LiveJournal 2014 | 74.84 | 57.86 | 33 / 42 |
| SMS 2013 | 70.28 | 49.00 | 34 / 42 |
| Twitter 2013 | 72.12 | 50.18 | 37 / 42 |
| Twitter 2014 | 70.96 | 55.47 | 32 / 42 |
| Twitter 2014 Sarcasm | 58.16 | 48.64 | 15 / 42 |

Table 3: Official results for Semeval 2014 test set. Reported scores are overall $F_1$ scores.

## 4 Conclusion

In this paper, we have described the submission of the SAP-RI team to the SemEval 2014 task 9. We showed that is possible to develop sentiment analysis systems via rapid prototyping with reasonable accuracy within a couple of days.

## Acknowledgement

The research is partially funded by the Economic Development Board and the National Research Foundation of Singapore.

## References


Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on Twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 36–44.

Albert Bifet, Geoffrey Holmes, Bernhard Pfahringer, and Ricard Gavalda. 2011. Detecting sentiment change in Twitter streaming data. *Journal of Machine LearningResearch - Proceedings Track*, 17:5–11.

Tao Chen and Min-Yen Kan. 2012. Creating a live, public short message service corpus: the NUS SMS corpus. *Language Resources and Evaluation*, pages 1–37.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177.

Bernhard J. Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. 2009. Twitter power: Tweets as electronic word of mouth. *J. Am. Soc Inf. Sci. Technol.*, 60(11):2169–2188.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Saif Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-Canada: Building the state-of-the-art in sentiment analysis of Tweets. In *Proceedings of the 7th International Workshop on Semantic Evaluation*, pages 321–327.

Preslav Nakov, Zornitsa Kozareva, Alan Ritter, Sara Rosenthal, Veselin Stoyanov, and Theresa Wilson. 2013. Semeval-2013 task 2: Sentiment analysis in Twitter. In *Proceedings of the 7th International Workshop on Semantic Evaluation*, pages 312–320.

Brendan O'Connor, Routledge Bryan R. Balasubramanyan, Ramnath, and Noah A. Smith. 2010. From Tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM '10)*, pages 122–129.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 380–390.

Alexander Pak and Patrick Paroubek. 2010. Twitter based system: Using Twitter to disambiguating sentiment ambiguous adjectives. In *Proceedings of the 8th International Workshop on Semantic Evaluation*, pages 436–439.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Oliver Grisel, Mathieu Blondel, Peter. Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Sara Rosenthal, Preslav Nakov, Alan Ritter, and Veselin Stoyanov. 2014. SemEval-2014 Task 9: Sentiment Analysis in Twitter. In Preslav Nakov and Torsten Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation*, SemEval '14, Dublin, Ireland.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP)*, pages 307–314.