How much does word sense disambiguation help in sentiment analysis of micropost data?

Chiraag Sumanth
PES Institute of Technology
Bangalore, India
chiraagsumanth@gmail.com

Diana Inkpen
University of Ottawa
Ottawa, Canada
diana.inkpen@uottawa.ca

Abstract

This short paper describes a sentiment analysis system for micro-post data that includes analysis of tweets from Twitter and Short Messaging Service (SMS) text messages. We discuss our system that makes use of Word Sense Disambiguation techniques in sentiment analysis at the message level, where the entire tweet or SMS text was analysed to determine its dominant sentiment. Previous work done in the area of Word Sense Disambiguation does not throw light on its influence on the analysis of social-media text and micropost data, which is what our work aims to achieve. Our experiments show that the use of Word Sense Disambiguation alone has resulted in an improved sentiment analysis system that outperforms systems built without incorporating Word Sense Disambiguation.

1 Introduction

Twitter is an online social networking and microblogging service that enables users to send and read short 140-character messages called ”tweets”. As of the first quarter of 2015, the microblogging service averaged at 236 million monthly active users. Worldwide over 350 billion SMS text messages are exchanged across the world's mobile networks every month, with over 15 percent of these messages being classified as commercial or marketing messages. The process of sentiment analysis involves text analytics, linguistics and accepted language processing to determine and dig subjective information from source materials. Sentiment analysis finds applications in various domains such as marketing, business and commerce (Jansen et al., 2009), healthcare (Chew and Eysenbach, 2010; Salathe and Khandelwal, 2011; Greaves et al., 2013), tourism and travel (Gonzalez-Rodriguez et al., 2014), and disaster management (Verma et al., 2011; Gao et al., 2011; Mandel et al., 2012).

One of the first problems that is encountered by any natural language processing system is that of lexical ambiguity, be it syntactic or semantic (Jurafsky and Martin, 2008). The resolution of a word's syntactic ambiguity has largely been solved in language processing by part-of-speech taggers which predict the syntactic category of words in text with high levels of accuracy. The problem is that words often have more than one meaning, sometimes fairly similar and sometimes completely different. The meaning of a word in a particular usage can only be determined by examining its context. **Word Sense Disambiguation (WSD)** is the process of identifying the sense of a polysemic word[1]. Different approaches to WSD (Mihalcea, 2010) include knowledge-based systems such as Lesk algorithm and adapted Lesk algorithm (Banerjee and Pederson, 2002), unsupervised corpus-based systems (Schutze, 1998; Ng, Wang, and Chan, 2003), and supervised corpus-based systems (Chklovski and Mihalcea, 2002).

Subjectivity Word Sense Disambiguation (SWSD) was shown to improve contextual opinion analysis by Akkaya et al. (2009). The authors state that SWSD is midway between pure dictionary classification and pure contextual interpretation. For SWSD, the context of the word is considered in order to perform the task, but the subjectivity is determined solely by the dictionary. A supervised learning approach was used, in which a different classifier was trained for each lexicon entry for which training data was present. Thus, they described their work as similar to targeted WSD, with two labels *Subjective* (S) and *Objective* (O). By applying SWSD to contextual polarity classification (positive/negative/neutral),

[1] As described in http://aclweb.org

Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015), pages 115–121,
Lisboa, Portugal, 17 September, 2015. ©2015 Association for Computational Linguistics.

they observed an accuracy improvement of 3 percentage points over the original classifier (Wilson et al., 2005a) calculated on the SenMPQA dataset. Additionally, Rentoumi et al. (2009) showed that WSD is valuable in polarity classification of sentences containing figurative expressions.

It should be noted that the above work did not focus on using WSD for social-media or micropost data, which is the primary focus area of our work.

Babelfy (Moro et al., 2014)[2] is a unified, multilingual, graph-based approach to Entity Linking and Word Sense Disambiguation based on a loose identification of candidate meanings coupled with a densest sub-graph heuristic which selects high-coherence semantic interpretations. We have used Babelfy for WSD in our work. Babelfy is based on the BabelNet 3.0 multilingual semantic network (Navigli and Ponzetto, 2012), and jointly performs WSD and entity linking in three steps:

- It associates with each vertex of the BabelNet semantic network, i.e., either concept or named entity, a semantic signature, that is, a set of related vertices. This is a preliminary step which needs to be performed only once, independently of the input text.

- Given an input text, it extracts all the linkable fragments from this text and, for each of them, lists the possible meanings according to the semantic network.

- It creates a graph-based semantic interpretation of the whole text by linking the candidate meanings of the extracted fragments using the previously-computed semantic signatures. It then extracts a dense sub-graph of this representation and selects the best candidate meaning for each fragment.

BabelNet 3.0, on which Babelfy is based, is obtained from the automatic integration of WordNet 3.0, Open Multilingual WordNet, Wikipedia, OmegaWiki, Wiktionary and Wikidata. We chose to use Babelfy for WSD as experiments on six gold-standard datasets show the state-of-the-art performance of Babelfy, as well as its robustness across languages. Its evaluation also demonstrates that Babelfy fares well both on long texts, such as those of the WSD tasks, and short and highly-ambiguous sentences, such as the ones in KORE50.[3]

2 Dataset

We used the Dataset from Conference on Semantic Evaluation Exercises (SemEval-2013) (Wilson et al., 2013)[4] for *Task 2: Sentiment Analysis in Twitter* and focused on sub-task B where the sentiment for the entire tweet/SMS was supposed to be determined. The organizers created and shared sentiment-labelled tweets for training, development, and testing. The task organizers also provided a second test dataset, composed of Short Message Service (SMS) messages. However, no SMS specific training data was provided or used. The datasets we used are described in Table 1.

Dataset	Positive	Negative	Neutral
Tweets			
Train	3,045 (37%)	1,209 (15%)	4,004 (48%)
Test	1,527 (41%)	601 (16%)	1,640 (43%)
SMS			
Test	492 (23%)	394 (19%)	1,208 (58%)

Table 1: Dataset Class Distribution.

The total number of annotated tweets in the training data is 8,258 tweets and in the testing data is 3,813 tweets. The total number of messages in the SMS testing data is 2,094 messages.

3 System Description

We will describe the system we have developed in the following sections.

3.1 Lexicons

Our system made use of a single lexical resource described below:

- SentiWordNet (Baccianella et al., 2010) is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity; that is, SentiWordNet contains positivity, negativity, and objectivity scores for each sense of a word, totally adding up to 1.0 for every sense of the word.

[2] http://babelfy.org/about

[3] http://www.yovisto.com/labs/ner-benchmarks/

[4] http://www.cs.york.ac.uk/semeval-2013/task2

3.2 Features

We used the tokenizer of the Carnegie Mellon University (CMU) Twitter NLP tool (Gimpel et al., 2011) to tokenize the training and testing data. We also performed more pre-processing such as stop-word removal and word stemming using the tools provided by the NLTK: the Natural Language Toolkit (Loper and Bird, 2002). Additionally, we used word segmentation for hashtags (starting with #) and user-ids (starting with @) re-inserted them after segmentation.

Each tweet or SMS text was represented as a vector made up of three features:

- For each term in the pre-processed text, retrieve the SentiWordNet scores for that sense matching the same sense of that term word, determined from Babelfy. This is not performed for terms that do not appear in SentiWordNet. These are the three features:
  - The total positive score for the entire text, determined by aggregating the SentiWordnet Positive (P) scores of the each sentiment for every term and normalized by dividing this by the total length of the text.
  - The total negative score for the entire text, determined by aggregating the SentiWordnet Negative (N) scores of the each sentiment for every term and normalized by dividing this by the total length of the text.
  - The total Neutral/Objective[5] score for the entire text, determined by aggregating the SentiWordnet Objective (O) scores of the each sentiment for every term and normalized by dividing this by the total length of the text.

4 Results

The initial phase of the system is unsupervised, where the unlabelled tweets and SMS text messages in the test dataset are pre-processed as described in the previous section and then subject to the following:

- Word Sense Disambiguation, of all possible terms in the text, using Babelfy.
- Matching the disambiguated word senses for each term with the Positive (P), Negative (N) and Objective/Neutral (O) scores from the matching sense of that term, using SentiWordNet. The total P, N and O scores for the text are calculated as described in the previous section.

The output of the above phase is the three-featured vector representation of each tweet or SMS text message.

We subsequently use supervision to make the system learn how to combine these three numeric features, representing each text, and reach a decision on the sentiment of that text. Thus, we repeat the above process and construct a three-featured vector (P, N and O scores) representation for each tweet present in the training dataset to be used by a supervised classifier for training.

This combined approach has the following advantages:

- Large amounts of unlabelled data can be processed and the three-featured vector representation for that dataset can be constructed without any supervision or training required.
- We use only three features (P, N and O scores) in the supervised training, and also do not use dataset-specific features such as bag of words, and therefore, the system should be easily adaptable to process other microposts datasets as well even if the topic words change in time (the so-called concept drift phenomenon).

We used supervised learning classifiers from Weka (Witten and Frank, 2005). As for the exact classifier, we used the Random Forest Decision Tree with their default settings. Random forests correct for decision trees' habit of over-fitting to their training set.

We decided to use the Random Forest over a Support Vector Machine (SVM), called SMO in Weka as the Random Forest outperformed the SMO model (default configuration in Weka) in both 10-fold cross validation of the training data, and also when used with the testing data. Random Forest has been previously shown to have outperformed SVM (Caruana and Niculescu-Mizil, 2006).

Table 2 below shows the overall accuracy for the baseline and our system, evaluated based on

[5] The SemEval organizers considered Neutral and Objective as equivalent in the dataset, which is why we have chosen to use them interchangeably here.

10-fold cross validation on the provided training data (contained only tweets but no SMS texts), using the Random Forest classifier. The baseline in Table 6 is the accuracy of a trivial classifier that puts everything in the most frequent class, which is Neutral/Objective for the training data (the ZeroR classifier in Weka).

System	Accuracy
Baseline	45.26
Our System	58.55

Table 2: Accuracies reported for 10-fold cross validation of training data.

The Precision, Recall and F-score metrics for the Twitter test data are shown in Table 3.

Class	Precision	Recall	F-Score
Positive	69.40	54.30	60.90
Negative	57.50	31.50	40.60
Neutral	60.00	81.30	69.10

Table 3: Results for Twitter test data, for each class.

The Precision, Recall and F-score metrics for the SMS test data are shown in Table 4.

Class	Precision	Recall	F-Score
Positive	52.60	62.80	57.30
Negative	67.50	30.40	41.90
Neutral	73.30	81.30	77.10

Table 4: Results for SMS test data, for each class.

Our main focus is to show whether Word Sense Disambiguation helps improve sentiment analysis of micropost data. Therefore, we have evaluated our system using only unigram lexicons and compared our results with that of the all-unigram-features results of the system developed by NRC-Canada (Mohammad et al., 2013), that was ranked first in the same task in the SemEval 2013 competition[6]. These unigram features included punctuation, upper-case words, POS tags, hashtags, unigram-only emotion and sentiment lexicons, emoticon detection, elongated words, and negation detection.

It may be noted that the NRC-Canada system did use several other bigram and n-gram features in their final, best-scoring submission such as word-ngrams, character-ngrams, token-clusters and multiple lexicons containing unigram, bigram, unigram-bigram pairs and bigram-bigram pairs, none of which we are using. It did not however feature the use of WSD.

In this work, we are not trying to show that our system is the best-scoring system in this task. Instead, we choose to only use unigram lexicons, and compared our results to that of the NRC-Canada system's reported score for all-unigram-features, and show the improvement observed over that score only, by using WSD for sentiment analysis.

Table 5 summarizes the results obtained by NRC-Canada for their system using all-unigram-features, and the results obtained with our system. The official metric used for evaluating system performance by the task organizers is average F-score for the positive and negative class.

Dataset	Tweets	SMS
Baseline 1 (Majority classifier)	29.19	19.03
Baseline 2 (First sense of correct POS)	34.65	29.75
NRC-Canada (All unigram features)	39.61	39.29
Our System	**50.75**	**49.60**

Table 5: Comparison of Average F-scores for positive/negative classes. All scores reported are for the test datasets

Table 5 also shows baseline results (Baseline 1) obtained by a majority classifier that always predicts the most frequent class as output. Since the final Average F-score is based only on the F-scores of positive and negative classes and not on neutral, the majority baseline shown, chose the most frequent class among positive and negative, which in this case was the positive class. The results shown in Baseline 2 are obtained for an similar system as ours, but in this case, we do not disambiguate word senses, and instead the reported SentiWordNet scores of first sense of the word for the right part-of-speech are chosen.

[6] We chose SemEval 2013 data and not data from the more recent editions of SemEval, because unigram-features-only score of the best scoring system (NRC-Canada) was reported in their SemEval 2013 submission. There has been no reported changes or improvements for the all-unigram-features only model in the recent editions. Additionally, the training data remained the same as SemEval 2013 for the recent editions as well.

It should be noted that we have only used three numeric-feature vectors to represent the data for training our system and no additional features such as unigram or n-grams, punctuation, token-clusters, upper-case words, elongated words, negation detection, emoticons or n-gram lexicons have been used. Using so few features has also helped determine that the considerable improvement in performance reported below can be primarily attributed only to WSD and the P, N and O scores that are determined from the SentiWordNet lexicon as a result of disambiguating the text, which then form the only three features in the vector used to represent the message. There are no other features used in our system that can claim to have contributed to the improved performance.

Therefore, we report an improvement of **11.14** percentage points for tweets and **10.31** percentage points for SMS text messages, over the all-unigram-features score of the NRC-Canada best-scoring system, when evaluated for the test dataset provided, despite our system not utilizing several other unigram features that were discussed above, but focussing only on the three WSD features instead.

5 Error Analysis

The results obtained reveal that the worst performing class as the Negative class. In both the cases of tweets and SMS text messages, the Precision and Recall for the Negative class is relatively lower than the same for the Positive and Neutral classes.

Error Analysis of the supervised classifier output revealed that the following may be the reasons:

1. Considerably lesser samples of negative tweets in training data (comprises only 15% of the training dataset). Therefore, the trained model maybe biased towards the more frequent classes, that is Positive and Neutral classes.

2. We have used SentiWordNet as the only lexical resource and no polarity or sentiment lexicons were used. Removal of such lexicons was reported to have the highest negative pact on performance (a loss in F-score of above 8.5 points for both tweets and SMS text) according to Mohammad et al. (2013)

3. We have not used word n-grams or character n-grams in our system as features and this was also reported to have a detrimental impact on performance (a loss in F-score of 7.25 points for tweets and 1.2 points for SMS text) according to (Mohammad et al., 2013)

4. Our system does not feature any negation-detection or encoding-detection, such as emoticons, punctuations, or upper-case letters which may characterize the sentiment of the entire text.

5. Accuracy of SentiWordNet sentiments and WSD of Babelfy[7] may have resulted in wrong sentiment scores being given and affected system performance.

It is important to note that these features have not been included into our current system as the objective of this work is to establish the primary contribution and influence of Word Sense disambiguation, without being aided by other features, in the improvement of sentiment analysis on social-media and micropost data. However, our future work will explore the addition of several other features to the current system, in addition to the existing WSD-aided features to further improve system performance.

6 Conclusion

We have presented our system that throws light on the positive influence that WSD can have when it comes to analyzing social-media and micropost data. We observe significant and considerable improvements obtained in sentiment analysis of micropost data such as tweets and SMS text messages, that can be primarily attributed only to WSD, when compared to systems developed without using WSD. Our approach, a combination of unsupervised and supervised phases, does not make use of any dataset-dependent features, it can be easily adapted to analyze other micropost datasets as well. It can also work well for future data. Since we are not using bag of words features, our system is not prone to performance degradation due to concept drift.

Acknowledgements

The authors would like to thank the Mitacs Globalink program for facilitating this collaborative work.

[7] Babelfy reported an F1-score of 84.6 on the SemEval 2007 WSD dataset. However this is not a micropost or social-media text dataset.

References

Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. 2009. Subjectivity word sense disambiguation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 190–199. Association for Computational Linguistics.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 2200–2204.

Satanjeev Banerjee and Ted Pedersen. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Computational linguistics and intelligent text processing*, pages 136–145. Springer.

Rich Caruana and Alexandru Niculescu-Mizil. 2006. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM.

Cynthia Chew and Gunther Eysenbach. 2010. Pandemics in the age of twitter: content analysis of tweets during the 2009 h1n1 outbreak. *PloS one*, 5(11):e14118.

Timothy Chklovski and Rada Mihalcea. 2002. Building a sense tagged corpus with open mind word expert. In *Proceedings of the ACL-02 workshop on Word sense disambiguation: recent successes and future directions-Volume 8*, pages 116–122. Association for Computational Linguistics.

Huiji Gao, Geoffrey Barbier, and Rebecca Goolsby. 2011. Harnessing the crowdsourcing power of social media for disaster relief. *IEEE Intelligent Systems*, (3):10–14.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics.

MR Gonzalez-Rodriguez, MR Martinez-Torres, and SL Toral. 2014. Monitoring travel-related information on social media through sentiment analysis. In *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, pages 636–641. IEEE Computer Society.

Felix Greaves, Daniel Ramirez-Cano, Christopher Millett, Ara Darzi, and Liam Donaldson. 2013. Harnessing the cloud of patient experience: using social media to detect poor quality healthcare. *BMJ quality & safety*, 22(3):251–255.

Bernard J Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. 2009. Twitter power: Tweets as electronic word of mouth. *Journal of the American society for information science and technology*, 60(11):2169–2188.

Daniel Jurafsky and James Martin. 2008. Speech and language processing: An introduction to speech recognition. *Computational Linguistics and Natural Language Processing. Prentice Hall.*

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, pages 63–70. Association for Computational Linguistics.

Benjamin Mandel, Aron Culotta, John Boulahanis, Danielle Stark, Bonnie Lewis, and Jeremy Rodrigue. 2012. A demographic analysis of online sentiment during hurricane irene. In *Proceedings of the Second Workshop on Language in Social Media*, pages 27–36. Association for Computational Linguistics.

Rada Mihalcea. 2010. Word sense disambiguation. In *Encyclopedia of Machine Learning*, pages 1027–1030. Springer.

Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 321–327.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Roberto Navigli and Simone Paolo Ponzetto. 2010. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 216–225. Association for Computational Linguistics.

Hwee Tou Ng, Bin Wang, and Yee Seng Chan. 2003. Exploiting parallel texts for word sense disambiguation: An empirical study. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 455–462. Association for Computational Linguistics.

Vassiliki Rentoumi, George Giannakopoulos, Vangelis Karkaletsis, and George A Vouros. 2009. Sentiment analysis of figurative language using a word sense disambiguation approach. In *RANLP*, pages 370–375.

Marcel Salathé and Shashank Khandelwal. 2011. Assessing vaccination sentiments with online social media: implications for infectious disease dynamics and control. *PLoS Comput Biol*, 7(10):e1002199.

Hinrich Schütze. 1998. Automatic word sense discrimination. *Computational linguistics*, 24(1):97–123.

Sudha Verma, Sarah Vieweg, William J Corvey, Leysia Palen, James H Martin, Martha Palmer, Aaron Schram, and Kenneth Mark Anderson. 2011. Natural language processing to the rescue? extracting" situational awareness" tweets during mass emergency. In *ICWSM*. Citeseer.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics.

Ian H Witten and Eibe Frank. 2005. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.