CrystalNest at SemEval-2017 Task 4: Using Sarcasm Detection for Enhancing Sentiment Classification and Quantification

Raj Kumar Gupta and **Yinping Yang***
Institute of High Performance Computing (IHPC)
Agency for Science, Technology and Research (A*STAR), Singapore
{*gupta-rk*, *yangyp*}@ihpc.a-star.edu.sg

Abstract

This paper describes a system developed for a shared sentiment analysis task and its subtasks organized by SemEval-2017. A key feature of our system is the embedded ability to detect sarcasm in order to enhance the performance of sentiment classification. We first constructed an affect-cognition-sociolinguistics sarcasm features model and trained a SVM-based classifier for detecting sarcastic expressions from general tweets. For sentiment prediction, we developed *CrystalNest*—a two-level cascade classification system using features combining sarcasm score derived from our sarcasm classifier, sentiment scores from Alchemy, NRC lexicon, n-grams, word embedding vectors, and part-of-speech features. We found that the sarcasm detection derived features consistently benefited key sentiment analysis evaluation metrics, in different degrees, across four subtasks A-D.

1 Introduction

Sentiment analysis, also known as opinion mining, is the study of the feelings and opinions from user-generated content. Sarcasm detection, though very related, is a different topic of interest. As a classification task, the primary objective of sentiment analysis is to determine if a message is positive, negative, or neutral. In contrast, the objective of sarcasm detection is to determine if a message is sarcastic or not sarcastic.

To illustrate, let us look at two short text examples. Example 1 expresses a positive sentiment which has a slight mixed feeling, but it is not sarcastic. A very similar-looking Example 2 is sarcastic, and its underlying sentiment is negative.

Ex 1. Love my new phone! Only that the battery runs out very fast.

Ex 2. Love my new phone that runs out battery so fast!

In computational linguistics and NLP, detecting sarcasm is receiving increasing research interest (e.g., González-Ibáñez et al., 2011; Reyes et al., 2012; Liebrecht et al., 2013; Riloff et al., 2013; Rajadesingan et al., 2015; Bamman and Smith, 2015). While these studies recognized the linkage between sarcasm and sentiment and have proposed various techniques for detecting sarcasm, none directly studied the impact of sarcasm detection on sentiment analysis. Maynard and Greenwood (2014) is among the first to explore how to use sarcasm-related information to improve sentiment analysis. They proposed a rule-based method involving five rules such as using "#sarcasm" to flip a sentiment from positive to negative. However, their evaluation was performed on a relatively small test dataset of 400 tweets.

We believe that sentiment analysis systems will benefit from a systematically embedded ability to detect sarcasm. In the following, we describe our approach and present supportive findings evaluated on a large set of test data provided by SemEval-2017 Task 4 (Rosenthal et al., 2017).

2 Sarcasm Detection: An Affect-Cognition-Sociolinguistics (ACS) Feature Model

In order to capture discriminative and explainable sarcasm features, we sought to design a feature model based on review and synthesis across related studies such as natural language processing, linguistics, psychology, speech and communication, as well as neuroscience. Our

* Both authors contributed to this research equally. For correspondence, please contact yangyp@ihpc.a-star.edu.sg.

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 617–624,
Vancouver, Canada, August 3 - 4, 2017.

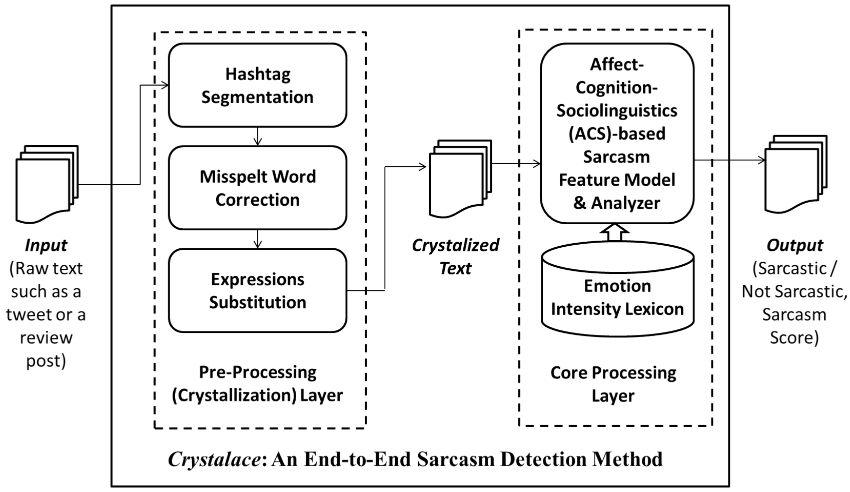

Figure 1: The Key Components of the *Crystalace* Sarcasm Detection Method

model characterizes sarcasm with three key feature groups: affect-related, cognition-related, and sociolinguistics-related features.

Figure 1 presents an overview of the proposed sarcasm detection method that we name it as "Crystalace". *Crystalace* will subsequently produce a key feature, i.e. sarcasm score, for the final *CrystalNest* sentiment analysis system. *Crystalace*'s core processing layer is the affect-cognition-sociolinguistics sarcasm feature model (sections 2.1-2.4). *Crystalace* also includes a supporting layer that pre-processes raw text into crystallized text (section 2.5) for effective feature extraction.

2.1 Affect-related features

A fundamental understanding of sarcasm is that it involves a negative emotional connotation through a seemingly positive expression (Brant, 2012). Riloff et al. (2013) suggested that *count* of positive and negative words, *location* and *order* of positive words and negative words are useful features in sarcasm detection. Rajadesingan et al. (2015) further used *strength* of positive words and negative words and found that strength-related features (e.g., count of very positive words in a tweet) are among top ten sarcasm features in their study.

In our model, beyond the valence and strength-related features, we propose to incorporate the intensity aspect of affective expressions. Conceptually, psychologists characterized emotion with two fundamental dimensions: the strength dimension (Osgood et al., 1957 called it "evaluation") in that an expression would have a positive or negative meaning that is strong, moderate or weak, and the intensity dimension (Shaver et al., 1987) which further concerns what Osgood et al. called motivational "potency" and physical "activity"[1]. With the intensity dimension, anger-based expressions (high in potency), for example, can be differentiated from sadness-based expressions (low in potency). Because sarcasm is featured with an underlying emotional connation (Brant, 2012), it is conceivable that expressers would tend to leverage seemingly positive emotions such as joy or gratitude words to implicate underlying negative mental experiences such as contempt or disapproval. Thus, in addition to the strength dimension, we explore capturing the emotional intensity variances to further differentiate sarcastic from non-sarcastic expressions.

Other than using words, Twitter users often use special punctuations to highlight their affective experiences, which can be useful cues to sarcasm. For example, users tend to capitalize certain letters to express strong feelings. Others may also use repetitive exclamations marks "!!!". Therefore, we consider these special punctuations as affect-related features. Lastly, we consider percentage of first-persons singular pronouns (I, me, mine etc.) as a feature as research in linguistic psychology has indicated that such words give an expresser power to make an emotional connection with the audience (Cohen, 2014).

2.2 Cognition-related features

Besides affect, sarcasm is also significantly associated with cognitive processes. As Haiman (1998) puts it, what is essential to sarcasm is that it is

[1]It is worth noting that other psychologists (e.g., Russell, 1980; Plutchik, 1980; Mehrabian, 1980) have also proposed other emotion dimensions.

"overt irony intentionally used by the speaker as a form of verbal aggression". Neuropsychology studies also indicated that damage of certain cognitive functions in the brain harms people's ability in recognizing sarcasm (Shamay-Tsoory et al., 2005; Davis et al., 2016). Because sarcasm is intentional, there is a degree of deliberation in order to construct sarcasm. Thus, if a sarcastic tweet is produced, the tweet is probably manifested with a high degree of lexical complexity which is also likely constructed by a high cognitive complexity individual. Conversely, a low cognitive complexity individual would tend to be more straightforward to communicate their feelings.

In linguistics, certain words have been found to reveal "depth of thinking" (Tausczik and Pennebaker, 2009). These include cognitive processes words (e.g., *because*), conjunctions (e.g., *although*), prepositions (e.g., *to*) and words greater than six letters. In addition, psycholinguistic analysis of tweets has suggested that a well-prepared and constructed tweet is correlated with higher lexical density, which is marked by information-carrying words (Hu et al., 2013). Therefore, we include nouns, negation, verbs, adjectives, numbers, and quantifiers which are information-carrying words in this feature category.

2.3 Sociolinguistics-related features

In verbal communication, average pitch, pitch slop, and laughter or responses to questions have been found to be prosodic cues to sarcasm utterances (Tepperman et al., 2006). In online digital platforms such as Twitter, users do not have facial and vocal cues at their disposal to communicate sarcastic expressions (Burgers, 2010). In consequence, they would find some alternative and "creative" ways to effectively express sarcasm cues as a hint to their intended audiences. Users would use hashtags to highlight a specific key phrase for easy search by others, use at-mentions to bring attention to a specific user, or use emoticons to provide cues to the underlying feelings. Therefore, we incorporate user-created hashtags, at-mentions, URLs and emoticons in our feature model.

2.4 Features Extraction

In total, our proposed sarcasm feature model includes a total of 82 features. The *affect-related* features include 50 valence-based features, strength-based features, intensity-based features and other indirect affective features. The *cognition-related* features include a total of 26 depth-of-thinking features (e.g., prep, conj). The *sociolinguistics-related* features refer to 6 Twitter-specific contextual cues features (e.g., #, @).

In order to capture the complementary benefits from different lexical sources, we used three lexicons, i.e., Opinion Lexicon[2] (Hu and Liu, 2004), SentiStrength Lookup Dictionary[3] (Thelwall et al., 2012), and our Emotion Intensity Lexicon[4], in conjunction with two linguistic sources, i.e., LIWC 2015[5] (Pennebaker et al., 2015) and TweetPOS[6] (Owoputi et al., 2013) to extract the relevant features.

Appendix A shows the full list of the 82 features, the feature codes and the respective linguistic resources/tools used for the features extraction.

2.5 Tweets Preprocessing

For supporting effective feature extraction, we designed a procedure to pre-process raw tweets. The first step is *hashtag segmentation* (Davidov et al., 2010), which involves tokenizing each hashtag such that the words can be more readily captured by existing lexical sources (e.g., *#shitnooneeversay* will be *shit no one ever say*). The second step is *misspelt word correction*, which converts words with more than two consecutive letters into those with two consecutive letters (e.g., *greaaat* will be *greaat*, *awwww* will be *aww*), such that intentionally misspelt words are standardized for the subsequent step. The third step is *expressions substi-*

[2]https://www.cs.uic.edu/∼liub/FBS/sentiment-analysis.html#lexicon

[3]http://sentistrength.wlv.ac.uk/

[4]No major sentiment or emotion lexicons developed to date cover the intensity dimension of emotions. Hence, we developed "Emotion Intensity (EI) Lexicon" for the purpose of more effectively distinguishing emotion-related words and phrases in different degrees of valence, strength and intensity. The EI Lexicon consists of 3,204 lexicon items including classic emotion-carrying English words, common social media slangs and emoticons, where each item is coded with a strength score as well as an intensity score in the range of [-3, -2, -1, 0, 1, 2, 3]. For example, items such as *excited*, *astonished* and *thrill* are coded as "3" (high-intensity, positive). Items such as *thank*, *cooperative*, *concern*, *:)* and *:d* are coded as "1" (low-intensity, positive). Items such as *sorry*, *agh* and *:/* are coded "-2" (medium-intensity, negative). Items such as *hate*, *resented* and *D:* are coded "-3" (high-intensity, negative). Words such as *great*, *haze*, *fulfill*, *sick* and *sleepy* are coded as "0" as they are related to emotions, but are not "genuine emotions" (Clore et al., 1987; Ortony et al., 1987). We will make this lexicon and its upgraded versions available for the research community.

[5]http://liwc.wpengine.com/

[6]http://www.cs.cmu.edu/ ark/TweetNLP/#pos

Method	Precision	Recall	F_1
Random Classifier	.22	.48	.30
N-grams Classifier	.54	.44	.48
Riloff et al. (2013)'s bootstrapped lexicon-based method	.62	.44	.51
Our proposed ACS model-based method (*Crystalace*)	**.52**	**.70**	**.60**

Table 1: Performance of Sarcasm Classification

tution. Even after the first two steps, many tweets could still contain a great variety of unusual expressions. Therefore, we constructed a mapped list of such expressions with more common words or phrases that carry a similar meaning, referencing Internet resources such as Urban Dictionary and Wikipedia. For example, *gonna* will be *going to*, *:/* will be *annoyed*, *aww* will be *sweet*, *classier* will be *excellent*, *rainy* will be *bad weather*, and *sneezing* will be *poor health*.

Note that we do not remove stop words, as removing stop words that helps in classic NLP tasks has been found to harm sentiment analysis performance (Saif et al., 2014).

2.6 Sarcasm Classifier

To train and evaluate our sarcasm classifier, we downloaded the annotated tweets dataset from Riloff et al. (2013), pre-processed the tweets, and trained a linear SVM classifier using our ACS-based features model. Similar to the final condition reported in Riloff et al. (2013), we also added unigrams and bigrams features to complement the theoretical features model. We then ran 10-fold cross validations to evaluate our method's performance. The results in Table 1 show that our ACS-based method obtained F_1-score of .60, which gained an additional .09 as compared to the best condition reported in Riloff et al.'s original study. Based on the results, we trained the final *Crystalace* sarcasm classifier using the full dataset.

3 System Description

Our sarcasm detection-enhanced sentiment analysis system, *CrystalNest*, is designed with five features groups and a cascade classifier with two levels of training. The following provides the development details.

3.1 Sarcasm and Sentiment Features

We used our *Crystalace* sarcasm classifier and Alchemy Language API[7] to form a two-dimensional feature vector. Alchemy Language is a component of the cognitive APIs offered on IBM Watson Developer Cloud. The first dimension of this feature vector contains the confidence score obtained using the sarcasm classifier and the second dimension contains the confidence score that has been obtained by calling Alchemy.

3.2 NRC SemEval-2015 English Twitter Lexicons Features

We also leveraged NRC SemEval-2015 English Twitter Sentiment Lexicons[8] which aims to capture the degree of the positiveness of a given word or phrase (Rosenthal et al., 2015) and a list of negator[9] words to extract a six-dimensional feature vector for each tweet. This feature vector contains the counts of positive, negative, neutral, negators words respectively, as well as maximum and minimum strengths of sentiment for a given tweet.

3.3 N-grams Features

N-grams are a common feature used for sentiment analysis. We extracted unigrams and bigrams from each tweet without removing stop words. To build the n-gram dictionary, we downloaded 25,000 general tweets using Twitter's Streaming API and extracted all possible unigrams and bigrams from those tweets. After extraction, we filtered these unigrams and bigrams based on their occurrences and removed all that appeared less than three times in our tweets dataset. We then used this n-gram dictionary to represent a tweet into the feature space where each of the feature dimensions represents the number of occurrences of that n-gram in the tweet.

3.4 Word Embedding Features

Word embedding has been used in recent Twitter sentiment analysis methods (Zhang et al., 2015; Rouvier and Favre, 2016) due to its ability to represent the semantic and syntactic meaning of

[7]https://www.ibm.com/watson/developercloud/alchemy-language.html

[8]http://saifmohammad.com/WebPages/lexicons.html

[9]http://dictionary.cambridge.org/grammar/british-grammar/questions-and-negative-sentences/negation and https://www.grammarly.com/handbook/sentences/negatives/1/negatives/

the word into a low-dimensional feature vector. Here, we used Gensim[10] based Sentence2Vec[11] to convert the tweets into 500-dimensional feature vectors. To train the word-embedding model, we downloaded approximately 8 million general tweets from Twitter using Twitter Streaming API.

3.5 Tweet Part-of-Speech (POS) Features

Lastly, we extracted 25-dimensional part-of-speech (Owoputi et al., 2013) features for each tweet *without* any preprocessing, as the TweetPOS tool has been specially designed to capture tweets-specific linguistic elements. These features help to capture cues such as tweets-specific linguistic counts, punctuation, as well as conversational markers including hashtags, at-mentions, emoticons and URLs.

3.6 Cascade Sentiment Classifier

For our final system, we used a cascade classification approach to predict the sentiment outcome. Before extracting the features, tweets are preprocessed as described in Section 2.5. For each of the five feature groups described in sections 3.1-3.5, we used linear SVM to train three different classifiers using one-against-all approach for positive, negative and neutral classes. For each of these classifiers (first-level classification), we used SemEval-2013 training data for training and SemEval-2016 and SemEval-2017 test tweets for final evaluation.

After obtaining the outputs from all three classifiers of each feature group, we formed a 15-dimensional feature vector and used Naive Bayes classifier to train the final classifier. In this final classifier (second-level classification), we used SemEval-2016 test data for training[12] and SemEval-2017 test data for final evaluation.

For topic-based tweet quantification subtask D, we calibrated *CrystalNest* using a dynamic base-sentiment selection approach as there was no clear prior knowledge to determine if topic-specific information would be benefiting or harming the quantification performance. We first obtained two sets of sentiment scores (*sentiment_general* and *sentiment_topic*) by using Alchemy to process each individual tweet's sentiment score *with* and *without* using the specific topic information. Then when *sentiment_general* and *sentiment_topic* converged on the same polarity, we used the converged consensus. When *sentiment_general* and *sentiment_topic* produced conflicting polarity for a given tweet, we used the "majority voted" polarity from the other tweets under the same topic to assign the polarity to the particular tweet that received conflicting polarity values. Using this dynamic approach, we found the error terms were reduced as compared to those resulted from simply relying on any of the individual *sentiment_general* and *sentiment_topic* base sentiment features.

4 Results

We evaluated the proposed approach using the official test datasets provided by SemEval-2017 Task 4's subtasks A-D. Tables 2-4 summarize the results. For subtasks A & B, recall and F_1 scores are assessed as averaged scores according to the task organizers (see Rosenthal et al. 2017 for detailed discussion on the evaluation metrics).

System	***Recall*(ρ)**	F_1^{PN}	***Acc***
Subtask A Message Polarity Classification			
Alchemy	.589	.577	.586
Alchemy+Sarcasm	.591	.575	.581
CyrstalNest	**.619**	**.593**	**.629**
Subtask B Topic-based Two-point Scale Classification			
Alchemy	.657	.651	.719
Alchemy+Sarcasm	.820	.816	.821
CyrstalNest	**.827**	**.822**	**.827**

Table 2: *CrystalNest* Results for Subtasks A & B

System	**MAE**^{M}	**MAE**$^\mu$
Subtask C Topic-based Five-point Scale Classification		
Alchemy	.758	.591
Alchemy+Sarcasm	.760	.564
CyrstalNest	**.698**	**.571**

Table 3: *CrystalNest* Results for Subtask C (MAE is an error term; the lower MAE is, the better the system is)

System	**KLD**	**AE**	**RAE**
Subtask D Topic-based Two-point Scale Quantification			
Alchemy	.357	.270	1.718
Alchemy+Sarcasm	.061	.111	1.346
CyrstalNest	**.056**	**.104**	**1.202**

Table 4: *CrystalNest* Results for Subtask D (KLD, AE and RAE are error terms; the lower they are, the better the system is)

[10]https://github.com/RaRe-Technologies/gensim

[11]https://github.com/klb3713/sentence2vec

[12]Note that for all the above-mentioned system training, we used only the classic general message-level sentiment (subtask A) data. This could limit the effectiveness of the training, and we plan to expand with more training data for future system enhancement.

The test data provided by SemEval-2017 Task 4 is so far one of the largest annotated sentiment analysis test datasets. Subtask A consists of 12,284 annotated tweets, Subtasks B and D consist of 6,185 annotated tweets, and Subtask C consists of 12,379 annotated tweets. The results indicated that *CrystalNest* consistently benefited the performance more than the full-fledged, off-the-shelf sentiment analysis service offered by Alchemy. Furthermore, when we experimented with the subsystem combining only Alchemy and sarcasm features, the enhancements from sarcasm classifier over Alchemy's base sentiment features were also found in subtasks A, B and D, in particular in the two two-point subtasks B and D.

In comparison with other participating systems, *CrystalNest* obtained relatively good rankings in subtask A (#18 out of 37 systems), subtask B (#9 out of 23), subtask C (#6 out of 15) and subtask D (#4 out of 15).

5 Conclusion

This paper described a new sentiment analysis system featuring a sarcasm detection classifier in conjunction with other complementary features derived from Alchemy, NRC sentiment lexicon, n-grams, word embedding vectors, and part-of-speech features. The evaluation results using sentiment analysis subtasks A-D test data provided initial evidence on the value of embedding sarcasm detection in sentiment analysis systems. For future work, we plan to explore deep learning methods and conduct more experiments to further augment the system performance.

Acknowledgment

This research is supported by the Social Technologies+ Programme funded by A*STAR Joint Council Office. We thank Tong Joo Chuan for the encouragement to pursue this research. The authors are grateful to Faith Tng Hui En and Tng Tai Hou for proofreading assistance and to anonymous reviewers for providing constructive comments that helped to improve this paper.

References

David Bamman and Noah Smith. 2015. Contextualized sarcasm detection on twitter. *International Conference on Weblogs and Social Media* pages 574–577.

William Brant. 2012. Critique of sarcastic reason: The epistemology of the cognitive neurological ability called 'theory-of-mind' and deceptive reasoning. *Südwestdeutscher Verlag für Hochschulschriften* .

Christian Frederik Burgers. 2010. Verbal irony: Use and effects in written discourse. *Ipskamp, UB Nijmegen, The Netherlands* .

Gerald L. Clore, Andrew Ortony, and Mark A. Foss. 1987. The psychological foundations of the affective lexicon. *Journal of Personality and Social Psychology* 53(4):751–766.

Georgy Cohen. 2014. The power of the first-person perspective. *http://meetcontent.com/blog/power-first-person-perspective/* .

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. *COLING* pages 241–249.

Cameron L. Davis, Kenichi Oishi, Andreia V. Faria, John Hsu, Yessenia Gomez, Susumu Mori, and Argye E. Hillis. 2016. White matter tracts critical for recognition of sarcasm. *Neurocase* 22(1):22–29.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: A closer look. *ACL: HLT* pages 581–586.

John Haiman. 1998. Talk is cheap: Sarcasm, alienation and the evolution of language. *Oxford University Press* page 20.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. *ACM SIGKDD* pages 168–177.

Yuheng Hu, Kartik Talamadupula, and Subbarao Kambhampati. 2013. Dude, srsly?: The surprisingly formal nature of twitter's language. *ICWSM* pages 244–253.

Christine Liebrecht, Florian Kunneman, and Antal Van Den Bosch. 2013. The perfect solution for detecting sarcasm in tweets #not. *Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* pages 29–37.

Diana Maynard and Mark A. Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. *LREC* pages 4238–4243.

Albert Mehrabian. 1980. Basic dimensions for a general psychological theory: Implications for personality, social, environmental, and developmental studies. *Oelgeschlager, Gunn & Hain* pages 39–53.

Andrew Ortony, Gerald L. Clore, and Mark A. Foss. 1987. The referential structure of the affective lexiconn. *Cognitive Science* 11(3):341–364.

Charles Egerton Osgood, George J. Suci, and Percy H. Tannenbaum. 1957. The measurement of meaning. *University of Illinois Press* .

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. *NAACL: HLT* pages 380–390.

James W. Pennebaker, Roger J. Booth, Ryan L. Boyd, and Martha E. Francis. 2015. Linguistic inquiry and word count: LIWC2015. *Austin, TX: Pennebaker Conglomerates* .

Robert Plutchik. 1980. Emotion: A psychoevolutionary synthesis. *New York: Harper and Row* .

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm detection on twitter: A behavioral modeling approach. *ACM WSDM* pages 97–106.

Antonio Reyes, Paolo Rosso, and Davide Buscaldi. 2012. From humor recognition to irony detection: The figurative language of social media. *Data & Knowledge Engineering* pages 1–12.

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. Sarcasm as contrast between a positive sentiment and negative situation. *Empirical Methods on Natural Language Processing* pages 704–714.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. *SemEval* .

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif M Mohammad, Alan Ritter, and Veselin Stoyanov. 2015. SemEval-2015 task 10: Sentiment analysis in twitter. *SemEval* pages 451–463.

Mickael Rouvier and Benoit Favre. 2016. SENSEI-LIF at SemEval-2016 task 4: Polarity embedding fusion for robust sentiment analysis. *SemEval* pages 202–208.

James A Russell. 1980. A circumplex model of affect. *Journal of Personality and Social Psychology* 39(6):1161–1178.

Hassan Saif, Miriam Fernndez, Yulan He, and Harith Alani. 2014. On stopwords, filtering and data sparsity for sentiment analysis of twitter. *LREC* pages 810–817.

Simone G. Shamay-Tsoory, Rachel Tomer, and Judith Aharon-Peretz. 2005. The neuroanatomical basis of understanding sarcasm and its relationship to social cognition. *Neuropsychology* pages 288–300.

Phillip Shaver, Judith Schwartz, Donald Kirson, and Cary O'Connor. 1987. Emotion knowledge: Further exploration of a prototype approach. *Journal of Personality and Social Psychology* 52(6):1061–1086.

Yla R. Tausczik and James W. Pennebaker. 2009. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology* 29(1):24–54.

Joseph Tepperman, David R. Traum, and Shrikanth Narayanan. 2006. yeah right: sarcasm recognition for spoken dialogue systems. *INTERSPEECH* pages 1838–1841.

Mike Thelwall, Kevan Buckley, and Georgios Paltoglou. 2012. Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology* 63(1):163–173.

Zhihua Zhang, Guoshun Wu, and Man Lan. 2015. Ecnu: Multi-level sentiment analysis on twitter usning traditional linguistic features and word embedding features. *SemEval* pages 561–567.

Appendix A Full List of Features in the Affect-Cognition-Sociolinguistics Sarcasm Feature Model

Features (example words)	**Feature codes**	**Extraction source/tool**
Affect-related Features (50)		
Count of +ive words (advanced, foolproof)	$pcountOL$	Opinion Lexicon
Count of −ive words (crashed, drunken)	$ncountOL$	
Count of both +ive and −ive words	$pncountOL$	
Starting position of first positive word (-1 if no positive word)	$pstartOL$	
Starting position of first negative word (-1 if no positive word)	$nstartOL$	
Order of the +ive and −ive words (1 if +ive words appear before−ive; -1 otherwise. 0 if no +ive/−ive words)	$pnorderOL$	
Count of positive words (2,3,4 scored) (care, bff)	$pcountSS$	SentiStrength Lookup Dictionary
Count of negative words (-2,-3,-4 scored) (dizzy, provoke)	$ncountSS$	
Count of both positive and negative words	$pncountSS$	
Starting position of first positive word	$pstartSS$	
Starting position of first negative word	$nstartSS$	
Order of the position of the positive and negative words	$pnorderSS$	
Count of 4-scored words (loving, magnific* [*: all words starting with magnific])	$pos4SS$	
Count of 3-scored words (awesome, fantastic, great, wow*, joy*)	$strengthp3SS$	
Count of 2-scored words (fun, glad, thank, nice*, brillian*)	$strengthp2SS$	
Count of 1-scored words (ok, peace*)	$strengthp1SS$	
Count of -1-scored words (dark, lost)	$strengthn1SS$	
Count of -2-scored words (against, aloof)	$strengthn2SS$	
Count of -3-scored words (envy*, foe*)	$strengthn3SS$	
Count of -4-scored words (cry, fear)	$strengthn4SS$	
Absolute value of highest positive strength score of words (e.g., 3 is returned if a tweet contains "excitement" and "amused", which have SentiStrength scores of 3 and 2 respectively)	$maxpstrengthSS$	
Absolute value of lowest negative strength score of words (e.g., 4 is returned if a tweet contains "anguish" and "alone", which have SentiStrength scores of -4 and -2 respectively)	$minnstrengthSS$	

Appendix A Full List of Features in the Affect-Cognition-Sociolinguistics Sarcasm Feature Model (continued...)

Features (example words)	Feature codes	Extraction source/tool
Affect-related Features (50) (continued...)		
Count of positive words (feeling-high, heartening, aww, =))	$pcountEI$	Emotion Intensity Lexicon
Count of negative words (uncared-for, weird, agh, :/)	$ncountEI$	
Count of both positive and negative words	$pncountEI$	
Starting position of first positive word	$pstartEI$	
Starting position of first negative word	$nstartEI$	
Order of the position of the positive and negative words	$pnorderEI$	
Count of 3-scored strength words (love, awesome)	$strengthp3EI$	
Count of 2-scored strength words (lucky, surprising)	$strengthp2EI$	
Count of 1-scored strength words (compassion, curious)	$strengthp1EI$	
Count of 0-scored strength words (refreshed, sleepy)	$strength0EI$	
Count of -1-scored strength words (nervous, sorrow)	$strengthn1EI$	
Count of -2-scored strength words (tense, bitter)	$strengthn2EI$	
Count of -3-scored strength words (woesome, hating)	$strengthn3EI$	
Absolute value of highest positive score of strength words	$maxpstrengthEI$	
Absolute value of highest negative score of strength words	$maxnstrengthEI$	
Count of 3-scored intensity words (excited, astonished, thrill)	$intensityp3EI$	
Count of 2-scored intensity words (love, awesome, glad, fun,:P,=D)	$intensityp2EI$	
Count of 1-scored intensity words (thank, cooperative, concern, :), :d)	$intensityp1EI$	
Count of 0-scored intensity words (great, haze, fulfill, sick, sleepy)	$intensity0EI$	
Count of -1-scored intensity words (anger, annoyed)	$intensityn1EI$	
Count of -2-scored intensity words (sorry, agh, :/)	$intensityn2EI$	
Count of -3-scored intensity words (hate, resented, D:)	$intensityn3EI$	
Absolute value of highest positive score of intensity words	$maxpintensityEI$	
Absolute value of lowest negative score of intensity words	$minnintensityEI$	
Percentage of uppercase characters	$uppcase$	LIWC2015
Percentage of question marks (?)	$qmark$	
Percentage of exclamation marks (!)	$exclamark$	
Percentage of first persons singular (I, me, mine)	i	
Cognition-related Features (26)		
Count of total words	WC	
Count of total characters	$charcount$	
Frequency of words greater than 6 letters	$sixltr$	
Percentage of negation words (no, never)	$negate$	
Percentage of certainty words	$certain$	
Percentage of preposition words	$prep$	
Percentage of conjunction words	$conj$	
Count of common nouns (books, someone)	N	TweetPOS
Count of pronoun (personal/WH; not possessive)	O	
Count of nominal + possessive words (books', someone's)	S	
Count of proper nouns (lebron, usa, iPad)	ˆ	
Count of proper nouns + possessive (America's)	Z	
Count of nominal _ verbal (I'm), verbal + nominal (let's)	L	
Count of proper noun + verbal (Mark'll)	M	
Count of verbs incl. copula and auxiliaries (might, ought, couldn't, is, eats)	V	
Count of adjectives (good, fav, lil)	A	
Count of adverbs (2, i.e., too)	R	
Count of interjections (lol, haha, FTW, yea, right)	!	
Count of determiner words (the, the, its, it's)	D	
Count of pre- or postpositions or subordinating conjunction (while, to, for, 2[to], 4[for])	P	
Count of coordinating conjunctions (and, n, &, +, BUT)	&	
Count of verb particles (out, off, Up, UP)	T	
Count of existential there, predeterminers (both)	X	
Count of existential there, predeterminers, verbal (there's, all's)	Y	
Count of numerals (2010, four, 9:30)	$	
Count of punctuations (#,$,(,))	,	
Sociolinguistics-related Features (6)		
Count of hashtags (#acl)	#	
Count of at-mentions (@BarackObama)	@	
Count of discourse markers (RT @user : hello)	∼	
Count of URLs or email address (http://t.co/rsxZxhnU)	U	
Count of emoticons (:) :b (: <3 o_O)	E	
Count of other abbreviations, foreign words etc. (ily (I love you) wby (what about you) 's –>awesome...I'm)	G	