Sarcasm as Contrast between a Positive Sentiment and Negative Situation

Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva,
Nathan Gilbert, Ruihong Huang
School Of Computing
University of Utah
Salt Lake City, UT 84112
{riloff,asheq,alnds,ngilbert,huangrh}@cs.utah.edu, prafulla.surve@gmail.com

Abstract

A common form of sarcasm on Twitter consists of a positive sentiment contrasted with a negative situation. For example, many sarcastic tweets include a positive sentiment, such as "love" or "enjoy", followed by an expression that describes an undesirable activity or state (e.g., "taking exams" or "being ignored"). We have developed a sarcasm recognizer to identify this type of sarcasm in tweets. We present a novel bootstrapping algorithm that automatically learns lists of positive sentiment phrases and negative situation phrases from sarcastic tweets. We show that identifying contrasting contexts using the phrases learned through bootstrapping yields improved recall for sarcasm recognition.

1 Introduction

Sarcasm is generally characterized as ironic or satirical wit that is intended to insult, mock, or amuse. Sarcasm can be manifested in many different ways, but recognizing sarcasm is important for natural language processing to avoid misinterpreting sarcastic statements as literal. For example, sentiment analysis can be easily misled by the presence of words that have a strong polarity but are used sarcastically, which means that the opposite polarity was intended. Consider the following tweet on Twitter, which includes the words "yay" and "thrilled" but actually expresses a negative sentiment: "*yay! it's a holiday weekend and i'm on call for work! couldn't be more thrilled! #sarcasm.*" In this case, the hashtag #sarcasm reveals the intended sarcasm, but we don't always have the benefit of an explicit sarcasm label.

In the realm of Twitter, we observed that many sarcastic tweets have a common structure that creates a positive/negative contrast between a sentiment and a situation. Specifically, sarcastic tweets often express a positive sentiment in reference to a negative activity or state. For example, consider the tweets below, where the positive sentiment terms are underlined and the negative activity/state terms are *italicized*.

(a) Oh how I love *being ignored*. #sarcasm

(b) Thoroughly enjoyed *shoveling the driveway* today! :) #sarcasm

(c) Absolutely adore it when *my bus is late* #sarcasm

(d) I'm so pleased mom *woke me up* with vacuuming my room this morning. :) #sarcasm

The sarcasm in these tweets arises from the juxtaposition of a positive sentiment word (e.g., love, enjoyed, adore, pleased) with a negative activity or state (e.g., being ignored, bus is late, shoveling, and being woken up).

The goal of our research is to identify sarcasm that arises from the contrast between a positive sentiment referring to a negative situation. A key challenge is to automatically recognize the stereotypically negative "situations", which are activities and states that most people consider to be unenjoyable or undesirable. For example, stereotypically unenjoyable activities include going to the dentist, taking an exam, and having to work on holidays. Stereotypically undesirable states include being ignored, having no friends, and feeling sick. People recognize

Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 704–714,
Seattle, Washington, USA, 18-21 October 2013. ©2013 Association for Computational Linguistics

these situations as being negative through cultural norms and stereotypes, so they are rarely accompanied by an explicit negative sentiment. For example, *"I feel sick"* is universally understood to be a negative situation, even without an explicit expression of negative sentiment. Consequently, we must learn to recognize phrases that correspond to stereotypically negative situations.

We present a bootstrapping algorithm that automatically learns phrases corresponding to positive sentiments and phrases corresponding to negative situations. We use tweets that contain a sarcasm hashtag as positive instances for the learning process. The bootstrapping algorithm begins with a single seed word, "love", and a large set of sarcastic tweets. First, we learn *negative situation phrases* that follow a positive sentiment (initially, the seed word "love"). Second, we learn *positive sentiment phrases* that occur near a negative situation phrase. The bootstrapping process iterates, alternately learning new negative situations and new positive sentiment phrases. Finally, we use the learned lists of sentiment and situation phrases to recognize sarcasm in new tweets by identifying contexts that contain a positive sentiment in close proximity to a negative situation phrase.

2 Related Work

Researchers have investigated the use of lexical and syntactic features to recognize sarcasm in text. Kreuz and Caucci (2007) studied the role that different lexical factors play, such as interjections (e.g., "gee" or "gosh") and punctuation symbols (e.g., '?') in recognizing sarcasm in narratives. Lukin and Walker (2013) explored the potential of a bootstrapping method for sarcasm classification in social dialogue to learn lexical N-gram cues associated with sarcasm (e.g., "oh really", "I get it", "no way", etc.) as well as lexico-syntactic patterns.

In opinionated user posts, Carvalho et al. (2009) found oral or gestural expressions, represented using punctuation and other keyboard characters, to be more predictive of irony[1] in contrast to features representing structured linguistic knowledge in Portuguese. Filatova (2012) presented a detailed description of sarcasm corpus creation with sarcasm annotations of Amazon product reviews. Their annotations capture sarcasm both at the document level and the text utterance level. Tsur et al. (2010) presented a semi-supervised learning framework that exploits syntactic and pattern based features in sarcastic sentences of Amazon product reviews. They observed correlated sentiment words such as "yay!" or "great!" often occurring in their most useful patterns.

Davidov et al. (2010) used sarcastic tweets and sarcastic Amazon product reviews to train a sarcasm classifier with syntactic and pattern-based features. They examined whether tweets with a sarcasm hashtag are reliable enough indicators of sarcasm to be used as a gold standard for evaluation, but found that sarcasm hashtags are noisy and possibly biased towards the hardest form of sarcasm (where even humans have difficulty). González-Ibáñez et al. (2011) explored the usefulness of lexical and pragmatic features for sarcasm detection in tweets. They used sarcasm hashtags as gold labels. They found positive and negative emotions in tweets, determined through fixed word dictionaries, to have a strong correlation with sarcasm. Liebrecht et al. (2013) explored N-gram features from 1 to 3-grams to build a classifier to recognize sarcasm in Dutch tweets. They made an interesting observation from their most effective N-gram features that people tend to be more sarcastic towards specific topics such as school, homework, weather, returning from vacation, public transport, the church, the dentist, etc. This observation has some overlap with our observation that stereotypically negative situations often occur in sarcasm.

The cues for recognizing sarcasm may come from a variety of sources. There exists a line of work that tries to identify facial and vocal cues in speech (e.g., (Gina M. Caucci, 2012; Rankin et al., 2009)). Cheang and Pell (2009) and Cheang and Pell (2008) performed studies to identify acoustic cues in sarcastic utterances by analyzing speech features such as speech rate, mean amplitude, amplitude range, etc. Tepperman et al. (2006) worked on sarcasm recognition in spoken dialogue using prosodic and spectral cues (e.g., average pitch, pitch slope, etc.) as well as contextual cues (e.g., laughter or response to questions) as features.

[1] They adopted the term 'irony' instead of 'sarcasm' to refer to the case when a word or expression with prior positive polarity is figuratively used to express a negative opinion.

While some of the previous work has identified specific expressions that correlate with sarcasm, none has tried to identify contrast between positive sentiments and negative situations. The novel contributions of our work include explicitly recognizing contexts that contrast a positive sentiment with a negative activity or state, as well as a bootstrapped learning framework to automatically acquire positive sentiment and negative situation phrases.

3 Bootstrapped Learning of Positive Sentiments and Negative Situations

Sarcasm is often defined in terms of contrast or "saying the opposite of what you mean". Our work focuses on one specific type of contrast that is common on Twitter: the expression of a positive sentiment (e.g., "love" or "enjoy") in reference to a negative activity or state (e.g., "taking an exam" or "being ignored"). Our goal is to create a sarcasm classifier for tweets that explicitly recognizes contexts that contain a positive sentiment contrasted with a negative situation.

Our approach learns rich phrasal lexicons of positive sentiments and negative situations using only the seed word "love" and a collection of sarcastic tweets as input. A key factor that makes the algorithm work is the presumption that if you find a positive sentiment or a negative situation in a sarcastic tweet, then you have found the source of the sarcasm. We further assume that the sarcasm probably arises from positive/negative contrast and we exploit syntactic structure to extract phrases that are likely to have contrasting polarity. Another key factor is that we focus specifically on tweets. The short nature of tweets limits the search space for the source of the sarcasm. The brevity of tweets also probably contributes to the prevalence of this relatively compact form of sarcasm.

3.1 Overview of the Learning Process

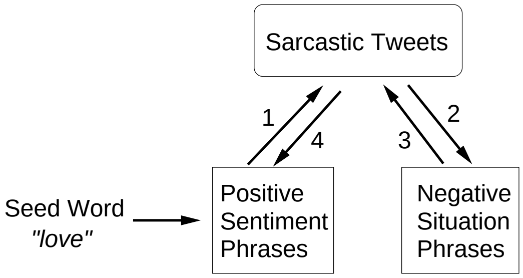

Figure 1: Bootstrapped Learning of Positive Sentiment and Negative Situation Phrases

Our bootstrapping algorithm operates on the assumption that many sarcastic tweets contain both a positive sentiment and a negative situation in close proximity, which is the source of the sarcasm.[2] Although sentiments and situations can be expressed in numerous ways, we focus on positive sentiments that are expressed as a verb phrase or as a predicative expression (predicate adjective or predicate nominal), and negative activities or states that can be a complement to a verb phrase. Ideally, we would like to parse the text and extract verb complement phrase structures, but tweets are often informally written and ungrammatical. Therefore we try to recognize these syntactic structures heuristically using only part-of-speech tags and proximity.

The learning process relies on an assumption that a positive sentiment verb phrase usually appears to the left of a negative situation phrase and in close proximity (usually, but not always, adjacent). Pictorially, we assume that many sarcastic tweets contain this structure:

[+ VERB PHRASE] [– SITUATION PHRASE]

This structural assumption drives our bootstrapping algorithm, which is illustrated in Figure 1. The bootstrapping process begins with a single seed word, "love", which seems to be the most common positive sentiment term in sarcastic tweets. Given a sarcastic tweet containing the word "love", our structural assumption infers that "love" is probably followed by an expression that refers to a negative situation. So we harvest the n-grams that follow the word "love" as negative situation candidates. We select the best candidates using a scoring metric, and add them to a list of negative situation phrases.

Next, we exploit the structural assumption in the opposite direction. Given a sarcastic tweet that contains a negative situation phrase, we infer that the negative situation phrase is preceded by a positive sentiment. We harvest the n-grams that precede the negative situation phrases as positive sentiment candidates, score and select the best candidates, and

[2] Sarcasm can arise from a negative sentiment contrasted with a positive situation too, but our observation is that this is much less common, at least on Twitter.

add them to a list of positive sentiment phrases. The bootstrapping process then iterates, alternately learning more positive sentiment phrases and more negative situation phrases.

We also observed that positive sentiments are frequently expressed as predicative phrases (i.e., predicate adjectives and predicate nominals). For example: *"I'm taking calculus. It is awesome. #sarcasm"*. Wiegand et al. (2013) offered a related observation that adjectives occurring in predicate adjective constructions are more likely to convey subjectivity than adjectives occurring in non-predicative structures. Therefore we also include a step in the learning process to harvest predicative phrases that occur in close proximity to a negative situation phrase. In the following sections, we explain each step of the bootstrapping process in more detail.

3.2 Bootstrapping Data

For the learning process, we used Twitter's streaming API to obtain a large set of tweets. We collected 35,000 tweets that contain the hashtag #sarcasm or #sarcastic to use as positive instances of sarcasm. We also collected 140,000 additional tweets from Twitter's random daily stream. We removed the tweets that contain a sarcasm hashtag, and considered the rest to be negative instances of sarcasm. Of course, there will be some sarcastic tweets that do not have a sarcasm hashtag, so the negative instances will contain some noise. But we expect that a very small percentage of these tweets will be sarcastic, so the noise should not be a major issue. There will also be noise in the positive instances because a sarcasm hashtag does not guarantee that there is sarcasm in the body of the tweet (e.g., the sarcastic content may be in a linked url, or in a prior tweet). But again, we expect the amount of noise to be relatively small.

Our tweet collection therefore contains a total of 175,000 tweets: 20% are labeled as sarcastic and 80% are labeled as not sarcastic. We applied CMU's part-of-speech tagger designed for tweets (Owoputi et al., 2013) to this data set.

3.3 Seeding

The bootstrapping process begins by initializing the positive sentiment lexicon with one seed word: *love*. We chose this seed because it seems to be the most common positive sentiment word in sarcastic tweets.

3.4 Learning Negative Situation Phrases

The first stage of bootstrapping learns new phrases that correspond to negative situations. The learning process consists of two steps: (1) harvesting candidate phrases, and (2) scoring and selecting the best candidates.

To collect candidate phrases for negative situations, we extract n-grams that follow a positive sentiment phrase in a sarcastic tweet. We extract every 1-gram, 2-gram, and 3-gram that occurs immediately after (on the right-hand side) of a positive sentiment phrase. As an example, consider the tweet in Figure 2, where "love" is the positive sentiment:

I love waiting forever for the doctor #sarcasm

Figure 2: Example Sarcastic Tweet

We extract three n-grams as candidate negative situation phrases:

waiting, waiting forever, waiting forever for

Next, we apply the part-of-speech (POS) tagger and filter the candidate list based on POS patterns so we only keep n-grams that have a desired syntactic structure. For negative situation phrases, our goal is to learn possible verb phrase (VP) complements that are themselves verb phrases because they should represent activities and states. So we require a candidate phrase to be either a unigram tagged as a verb (V) or the phrase must match one of 7 POS-based bigram patterns or 20 POS-based trigram patterns that we created to try to approximate the recognition of verbal complement structures. The 7 POS bigram patterns are: V+V, V+ADV, ADV+V, "to"+V, V+NOUN, V+PRO, V+ADJ. Note that we used a POS tagger designed for Twitter, which has a smaller set of POS tags than more traditional POS taggers. For example there is just a single V tag that covers all types of verbs. The V+V pattern will therefore capture negative situation phrases that consist of a present participle verb followed by a past participle verb, such as "being ignored" or "getting hit".[3] We also allow verb particles to match a V tag in our patterns. The remaining bigram patterns capture verb phrases that include a verb and adverb, an

[3] In some cases it may be more appropriate to consider the second verb to be an adjective, but in practice they were usually tagged as verbs.

infinitive form (e.g., "to clean"), a verb and noun phrase (e.g., "shoveling snow"), or a verb and adjective (e.g., "being alone"). We use some simple heuristics to try to ensure that we are at the end of an adjective or noun phrase (e.g., if the following word is tagged as an adjective or noun, then we assume we are *not* at the end).

The 20 POS trigram patterns are similar in nature and are designed to capture seven general types of verb phrases: verb and adverb mixtures, an infinitive VP that includes an adverb, a verb phrase followed by a noun phrase, a verb phrase followed by a prepositional phrase, a verb followed by an adjective phrase, or an infinitive VP followed by an adjective, noun, or pronoun.

Returning to Figure 2, only two of the n-grams match our POS patterns, so we are left with two candidate phrases for negative situations:

waiting, waiting forever

Next, we score each negative situation candidate by estimating the probability that a tweet is sarcastic given that it contains the candidate phrase following a positive sentiment phrase:

$$\frac{|\text{ follows(–candidate, +sentiment) \& sarcastic }|}{|\text{ follows(–candidate, +sentiment) }|}$$

We compute the number of times that the negative situation candidate immediately follows a positive sentiment in sarcastic tweets divided by the number of times that the candidate immediately follows a positive sentiment in all tweets. We discard phrases that have a frequency < 3 in the tweet collection since they are too sparse.

Finally, we rank the candidate phrases based on this probability, using their frequency as a secondary key in case of ties. The top 20 phrases with a probability $\geq .80$ are added to the negative situation phrase list.[4] When we add a phrase to the negative situation list, we immediately remove all other candidates that are subsumed by the selected phrase. For example, if we add the phrase "waiting", then the phrase "waiting forever" would be removed from the candidate list because it is subsumed by "waiting". This process reduces redundancy in the set of phrases that we add during each bootstrapping iteration. The bootstrapping process stops when no more candidate phrases pass the probability threshold.

3.5 Learning Positive Verb Phrases

The procedure for learning positive sentiment phrases is analogous. First, we collect phrases that potentially convey a positive sentiment by extracting n-grams that precede a negative situation phrase in a sarcastic tweet. To learn positive sentiment verb phrases, we extract every 1-gram and 2-gram that occurs immediately before (on the left-hand side of) a negative situation phrase.

Next, we apply the POS tagger and filter the n-grams using POS tag patterns so that we only keep n-grams that have a desired syntactic structure. Here our goal is to learn simple verb phrases (VPs) so we only retain n-grams that contain at least one verb and consist only of verbs and (optionally) adverbs. Finally, we score each candidate sentiment verb phrase by estimating the probability that a tweet is sarcastic given that it contains the candidate phrase preceding a negative situation phrase:

$$\frac{|\text{ precedes(+candidateVP,–situation) \& sarcastic }|}{|\text{ precedes(+candidateVP,–situation) }|}$$

3.6 Learning Positive Predicative Phrases

We also use the negative situation phrases to harvest predicative expressions (predicate adjective or predicate nominal structures) that occur nearby. Based on the same assumption that sarcasm often arises from the contrast between a positive sentiment and a negative situation, we identify tweets that contain a negative situation and a predicative expression in close proximity. We then assume that the predicative expression is likely to convey a positive sentiment.

To learn predicative expressions, we use 24 copular verbs from Wikipedia[5] and their inflections. We extract positive sentiment candidates by extracting 1-grams, 2-grams, and 3-grams that appear immediately after a copular verb and occur within 5 words of the negative situation phrase, on either side. This constraint only enforces proximity because predicative expressions often appear in a separate clause or sentence (e.g., *"It is just great that my iphone was stolen"* or *"My iphone was stolen. This is great."*)

[4] Fewer than 20 phrases will be learned if $<$ 20 phrases pass this threshold.

[5] http://en.wikipedia.org/wiki/List_of_English_copulae

We then apply POS patterns to identify n-grams that correspond to predicate adjective and predicate nominal phrases. For predicate adjectives, we retain ADJ and ADV+ADJ n-grams. We use a few heuristics to check that the adjective is not part of a noun phrase (e.g., we check that the following word is not a noun). For predicate nominals, we retain ADV+ADJ+N, DET+ADJ+N and ADJ+N n-grams. We excluded noun phrases consisting only of nouns because they rarely seemed to represent a sentiment. The sentiment in predicate nominals was usually conveyed by the adjective. We discard all candidates with frequency < 3 as being too sparse. Finally, we score each remaining candidate by estimating the probability that a tweet is sarcastic given that it contains the predicative expression near (within 5 words of) a negative situation phrase:

$$\frac{|\ \text{near(+candidatePRED,–situation) \& sarcastic}\ |}{|\ \text{near(+candidatePRED,–situation)}\ |}$$

We found that the diversity of positive sentiment verb phrases and predicative expressions is much lower than the diversity of negative situation phrases. As a result, we sort the candidates by their probability and conservatively add only the top 5 positive verb phrases and top 5 positive predicative expressions in each bootstrapping iteration. Both types of sentiment phrases must pass a probability threshold of $\geq .70$.

3.7 The Learned Phrase Lists

The bootstrapping process alternately learns positive sentiments and negative situations until no more phrases can be learned. In our experiments, we learned 26 positive sentiment verb phrases, 20 predicative expressions and 239 negative situation phrases.

Table 1 shows the first 15 positive verb phrases, the first 15 positive predicative expressions, and the first 40 negative situation phrases learned by the bootstrapping algorithm. Some of the negative situation phrases are not complete expressions, but it is clear that they will often match negative activities and states. For example, “getting yelled” was generated from sarcastic comments such as “I love getting yelled at”, “being home” occurred in tweets about “being home alone”, and “being told” is often being told what to do. Shorter phrases often outranked longer phrases because they are more general, and will therefore match more contexts. But an avenue for future work is to learn linguistic expressions that more precisely characterize specific negative situations.

Positive Verb Phrases (26): missed, loves, enjoy, cant wait, excited, wanted, can’t wait, get, appreciate, decided, loving, really like, looooove, just keeps, loveee, ...

Positive Predicative Expressions (20): great, so much fun, good, so happy, better, my favorite thing, cool, funny, nice, always fun, fun, awesome, the best feeling, amazing, happy, ...

Negative Situations (239): being ignored, being sick, waiting, feeling, waking up early, being woken, fighting, staying, writing, being home, cleaning, not getting, crying, sitting at home, being stuck, starting, being told, being left, getting ignored, being treated, doing homework, learning, getting up early, going to bed, getting sick, riding, being ditched, getting ditched, missing, not sleeping, not talking, trying, falling, walking home, getting yelled, being awake, being talked, taking care, doing nothing, wasting, ...

Table 1: Examples of Learned Phrases

4 Evaluation

4.1 Data

For evaluation purposes, we created a gold standard data set of manually annotated tweets. Even for people, it is not always easy to identify sarcasm in tweets because sarcasm often depends on conversational context that spans more than a single tweet. Extracting conversational threads from Twitter, and analyzing conversational exchanges, has its own challenges and is beyond the scope of this research. We focus on identifying sarcasm that is self-contained in one tweet and does not depend on prior conversational context.

We defined annotation guidelines that instructed human annotators to read isolated tweets and label a tweet as *sarcastic* if it contains comments judged to be sarcastic based solely on the content of that tweet. Tweets that do not contain sarcasm, or where potential sarcasm is unclear without seeing the prior conversational context, were labeled as *not sarcastic*. For example, a tweet such as *"Yes, I meant that sarcastically."* should be labeled as *not sarcastic* because the sarcastic content was (presumably) in a previous tweet. The guidelines did not contain any instructions that required positive/negative contrast to be present in the tweet, so all forms of sarcasm were considered to be positive examples.

To ensure that our evaluation data had a healthy mix of both sarcastic and non-sarcastic tweets, we collected 1,600 tweets with a sarcasm hashtag (#sarcasm or #sarcastic), and 1,600 tweets without these sarcasm hashtags from Twitter's random streaming API. When presenting the tweets to the annotators, the sarcasm hashtags were removed so the annotators had to judge whether a tweet was sarcastic or not without seeing those hashtags.

To ensure that we had high-quality annotations, three annotators were asked to annotate the same set of 200 tweets (100 sarcastic + 100 not sarcastic). We computed inter-annotator agreement (IAA) between each pair of annotators using Cohen's kappa (κ). The pairwise IAA scores were κ=0.80, κ=0.81, and κ=0.82. We then gave each annotator an additional 1,000 tweets to annotate, yielding a total of 3,200 annotated tweets. We used the first 200 tweets as our Tuning Set, and the remaining 3000 tweets as our Test Set.

Our annotators judged 742 of the 3,200 tweets (23%) to be sarcastic. Only 713 of the 1,600 tweets with sarcasm hashtags (45%) were judged to be sarcastic based on our annotation guidelines. There are several reasons why a tweet with a sarcasm hashtag might not have been judged to be sarcastic. Sarcasm may not be apparent without prior conversational context (i.e., multiple tweets), or the sarcastic content may be in a URL and not the tweet itself, or the tweet's content may not obviously be sarcastic without seeing the sarcasm hashtag (e.g., *"The most boring hockey game ever #sarcasm"*).

Of the 1,600 tweets in our data set that were obtained from the random stream and did not have a sarcasm hashtag, 29 (1.8%) were judged to be sarcastic based on our annotation guidelines.

4.2 Baselines

Overall, 693 of the 3,000 tweets in our Test Set were annotated as sarcastic, so a system that classifies every tweet as sarcastic will have 23% precision. To assess the difficulty of recognizing the sarcastic tweets in our data set, we evaluated a variety of baseline systems.

We created two baseline systems that use n-gram features with supervised machine learning to create a sarcasm classifier. We used the LIBSVM (Chang and Lin, 2011) library to train two support vector machine (SVM) classifiers: one with just unigram features and one with both unigrams and bigrams. The features had binary values indicating the presence or absence of each n-gram in a tweet. The classifiers were evaluated using 10-fold cross-validation. We used the RBF kernel, and the cost and gamma parameters were optimized for accuracy using unigram features and 10-fold cross-validation on our Tuning Set. The first two rows of Table 2 show the results for these SVM classifiers, which achieved F scores of 46-48%.

We also conducted experiments with existing sentiment and subjectivity lexicons to see whether they could be leveraged to recognize sarcasm. We experimented with three resources:

Liu05 : A positive and negative opinion lexicon from (Liu et al., 2005). This lexicon contains 2,007 positive sentiment words and 4,783 negative sentiment words.

MPQA05 : The MPQA Subjectivity Lexicon that is part of the OpinionFinder system (Wilson et al., 2005a; Wilson et al., 2005b). This lexicon contains 2,718 subjective words with positive polarity and 4,910 subjective words with negative polarity.

AFINN11 The AFINN sentiment lexicon designed for microblogs (Nielsen, 2011; Hansen et al., 2011) contains 2,477 manually labeled words and phrases with integer values ranging from -5 (negativity) to 5 (positivity). We considered all words with negative values to have negative polarity (1598 words), and all words with positive values to have positive polarity (879 words).

We performed four sets of experiments with each resource to see how beneficial existing sentiment

System	Recall	Precision	F score
Supervised SVM Classifiers			
1grams	.35	.64	.46
1+2grams	.39	.64	.48
Positive Sentiment Only			
Liu05	.77	.34	.47
MPQA05	**.78**	.30	.43
AFINN11	.75	.32	.44
Negative Sentiment Only			
Liu05	.26	.23	.24
MPQA05	.34	.24	.28
AFINN11	.24	.22	.23
Positive and Negative Sentiment, Unordered			
Liu05	.19	.37	.25
MPQA05	.27	.30	.29
AFINN11	.17	.30	.22
Positive and Negative Sentiment, Ordered			
Liu05	.09	.40	.14
MPQA05	.13	.30	.18
AFINN11	.09	.35	.14
Our Bootstrapped Lexicons			
Positive VPs	.28	.45	.35
Negative Situations	.29	.38	.33
Contrast(+VPs, –Situations), Unordered	.11	.56	.18
Contrast(+VPs, –Situations), Ordered	.09	**.70**	.15
& Contrast(+Preds, –Situations)	.13	.63	.22
Our Bootstrapped Lexicons ∪ SVM Classifier			
Contrast(+VPs, –Situations), Ordered	.42	.63	.50
& Contrast(+Preds, –Situations)	.44	.62	**.51**

Table 2: Experimental results on the test set

lexicons could be for sarcasm recognition in tweets. Since our hypothesis is that sarcasm often arises from the contrast between something positive and something negative, we systematically evaluated the positive and negative phrases individually, jointly, and jointly in a specific order (a positive phrase *followed by* a negative phrase).

First, we labeled a tweet as sarcastic if it contains any positive term in each resource. The *Positive Sentiment Only* section of Table 2 shows that all three sentiment lexicons achieved high recall (75-78%) but low precision (30-34%). Second, we labeled a tweet as sarcastic if it contains any negative term from each resource. The *Negative Sentiment Only* section of Table 2 shows that this approach yields much lower recall and also lower precision of 22-24%, which is what would be expected of a random classifier since 23% of the tweets are sarcastic. These results suggest that explicit negative sentiments are not generally indicative of sarcasm.

Third, we labeled a tweet as sarcastic if it contains both a positive sentiment term and a negative sentiment term, in any order. The *Positive and Negative Sentiment, Unordered* section of Table 2 shows that this approach yields low recall, indicating that relatively few sarcastic tweets contain both positive and negative sentiments, and low precision as well.

Fourth, we required the contrasting sentiments to occur in a specific order (the positive term must precede the negative term) and near each other (no more than 5 words apart). This criteria reflects our observation that positive sentiments often closely precede negative situations in sarcastic tweets, so we wanted to see if the same ordering tendency holds for negative sentiments. The *Positive and Negative Sentiment, Ordered* section of Table 2 shows that this ordering constraint further decreases recall and only slightly improves precision, if at all. Our hypothesis is that when positive and negative sentiments are expressed in the same tweet, they are referring to different things (e.g., different aspects of a product). Expressing positive and negative sentiments about the same thing would usually sound contradictory rather than sarcastic.

4.3 Evaluation of Bootstrapped Phrase Lists

The next set of experiments evaluates the effectiveness of the positive sentiment and negative situation phrases learned by our bootstrapping algorithm. The results are shown in the *Our Bootstrapped Lexicons* section of Table 2. For the sake of comparison with other sentiment resources, we first evaluated our positive sentiment verb phrases and negative situation phrases independently. Our positive verb phrases achieved much lower recall than the positive sentiment phrases in the other resources, but they had higher precision (45%). The low recall is undoubtedly because our bootstrapped lexicon is small and contains only verb phrases, while the other resources are much larger and contain terms with additional parts-of-speech, such as adjectives and nouns.

Despite its relatively small size, our list of negative situation phrases achieved 29% recall, which is comparable to the negative sentiments, but higher precision (38%).

Next, we classified a tweet as sarcastic if it contains both a positive verb phrase and a negative situation phrase from our bootstrapped lists, in any order. This approach produced low recall (11%) but higher precision (56%) than the sentiment lexicons. Finally, we enforced an ordering constraint so a tweet is labeled as sarcastic only if it contains a positive verb phrase that precedes a negative situation in close proximity (no more than 5 words apart). This ordering constraint further increased precision from 56% to 70%, with a decrease of only 2 points in recall. This precision gain supports our claim that this particular structure (positive verb phrase followed by a negative situation) is strongly indicative of sarcasm. Note that the same ordering constraint applied to a positive verb phrase followed by a negative *sentiment* produced much lower precision (at best 40% precision using the Liu05 lexicon). Contrasting a positive sentiment with a negative *situation* seems to be a key element of sarcasm.

In the last experiment, we added the positive predicative expressions and also labeled a tweet as sarcastic if a positive predicative appeared in close proximity to (within 5 words of) a negative situation. The positive predicatives improved recall to 13%, but decreased precision to 63%, which is comparable to the SVM classifiers.

4.4 A Hybrid Approach

Thus far, we have used the bootstrapped lexicons to recognize sarcasm by looking for phrases in our lists. We will refer to our approach as the Contrast method, which labels a tweet as sarcastic if it contains a positive sentiment phrase in close proximity to a negative situation phrase.

The Contrast method achieved 63% precision but with low recall (13%). The SVM classifier with unigram and bigram features achieved 64% precision with 39% recall. Since neither approach has high recall, we decided to see whether they are complementary and the Contrast method is finding sarcastic tweets that the SVM classifier overlooks.

In this hybrid approach, a tweet is labeled as sarcastic if either the SVM classifier or the Contrast method identifies it as sarcastic. This approach improves recall from 39% to 42% using the Contrast method with only positive verb phrases. Recall improves to 44% using the Contrast method with both positive verb phrases and predicative phrases. This hybrid approach has only a slight drop in precision, yielding an F score of 51%. This result shows that our bootstrapped phrase lists are recognizing sarcastic tweets that the SVM classifier misses.

Finally, we ran tests to see if the performance of the hybrid approach (Contrast $\cup$ SVM) is statistically significantly better than the performance of the SVM classifier alone. We used paired bootstrap significance testing as described in Berg-Kirkpatrick et al. (2012) by drawing 10^6 samples with repetition from the test set. These results showed that the Contrast $\cup$ SVM system is statistically significantly better than the SVM classifier at the $p < .01$ level (i.e., the null hypothesis was rejected with 99% confidence).

4.5 Analysis

To get a better sense of the strength and limitations of our approach, we manually inspected some of the tweets that were labeled as sarcastic using our bootstrapped phrase lists. Table 3 shows some of the sarcastic tweets found by the Contrast method but not by the SVM classifier.

i love *fighting* with the one i love
love *working* on my last day of summer
i enjoy tweeting [user] and *not getting* a reply
working during vacation is awesome .
can't wait *to wake* up early to babysit !

Table 3: Five sarcastic tweets found by the Contrast method but not the SVM

These tweets are good examples of a positive sentiment (love, enjoy, awesome, can't wait) contrasting with a negative situation. However, the negative situation phrases are not always as specific as they should be. For example, "working" was learned as a negative situation phrase because it is often negative when it follows a positive sentiment ("I love working..."). But the attached prepositional phrases ("on my last day of summer" and "during vacation") should ideally have been captured as well.

We also examined tweets that were incorrectly labeled as sarcastic by the Contrast method. Some false hits come from situations that are frequently negative but not always negative (e.g., some people genuinely like waking up early). However, most false hits were due to overly general negative situation phrases (e.g., "I love *working* there" was labeled as sarcastic). We believe that an important direction for future work will be to learn longer phrases that represent more specific situations.

5 Conclusions

Sarcasm is a complex and rich linguistic phenomenon. Our work identifies just one type of sarcasm that is common in tweets: contrast between a positive sentiment and negative situation. We presented a bootstrapped learning method to acquire lists of positive sentiment phrases and negative activities and states, and show that these lists can be used to recognize sarcastic tweets.

This work has only scratched the surface of possibilities for identifying sarcasm arising from positive/negative contrast. The phrases that we learned were limited to specific syntactic structures and we required the contrasting phrases to appear in a highly constrained context. We plan to explore methods for allowing more flexibility and for learning additional types of phrases and contrasting structures.

We also would like to explore new ways to identify stereotypically negative activities and states because we believe this type of world knowledge is essential to recognize many instances of sarcasm. For example, sarcasm often arises from a description of a negative event followed by a positive emotion but in a separate clause or sentence, such as: *"Going to the dentist for a root canal this afternoon. Yay, I can't wait."* Recognizing the intensity of the negativity may also be useful to distinguish strong contrast from weak contrast. Having knowledge about stereotypically undesirable activities and states could also be important for other natural language understanding tasks, such as text summarization and narrative plot analysis.

6 Acknowledgments

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center (DoI / NBC) contract number D12PC00285. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBE, or the U.S. Government.

References

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 995–1005.

Paula Carvalho, Luís Sarmento, Mário J. Silva, and Eugénio de Oliveira. 2009. Clues for detecting irony in user-generated contents: oh...!! it's "so easy" ;-). In *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, TSA 2009.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transac-*

tions on Intelligent Systems and Technology, 2:27:1–27:27.

Henry S. Cheang and Marc D. Pell. 2008. The sound of sarcasm. *Speech Commun.*, 50(5):366–381, May.

Henry S. Cheang and Marc D. Pell. 2009. Acoustic markers of sarcasm in cantonese and english. *The Journal of the Acoustical Society of America*, 126(3):1394–1405.

Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, CoNLL 2010.

Elena Filatova. 2012. Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.

Roger J. Kreuz Gina M. Caucci. 2012. Social and paralinguistic cues to sarcasm. *online 08/02/2012*, 25:1–22, February.

Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Lars Kai Hansen, Adam Arvidsson, Finn Arup Nielsen, Elanor Colleoni, and Michael Etter. 2011. Good friends, bad news - affect and virality in twitter. In *The 2011 International Workshop on Social Computing, Network, and Services (SocialComNet 2011)*.

Roger Kreuz and Gina Caucci. 2007. Lexical influences on the perception of sarcasm. In *Proceedings of the Workshop on Computational Approaches to Figurative Language*.

Christine Liebrecht, Florian Kunneman, and Antal Van den Bosch. 2013. The perfect solution for detecting sarcasm in tweets #not. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, WASSA 2013.

Bing Liu, Minqing Hu, and Junsheng Cheng. 2005. Opinion observer: Analyzing and comparing opinions on the web. In *Proceedings of the 14th International World Wide Web conference (WWW-2005)*.

Stephanie Lukin and Marilyn Walker. 2013. Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. In *Proceedings of the Workshop on Language Analysis in Social Media*.

Finn Arup Nielsen. 2011. A new anew: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on 'Making Sense of Microposts': Big things come in small packages (http://arxiv.org/abs/1103.2903)*.

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 *The 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2013)*.

Katherine P. Rankin, Andrea Salazar, Maria Luisa Gorno-Tempini, Marc Sollberger, Stephen M. Wilson, Danijela Pavlic, Christine M. Stanley, Shenly Glenn, Michael W. Weiner, and Bruce L. Miller. 2009. Detecting sarcasm from paralinguistic cues: Anatomic and cognitive correlates in neurodegenerative disease. *Neuroimage*, 47:2005–2015.

Joseph Tepperman, David Traum, and Shrikanth Narayanan. 2006. ”Yeah right”: Sarcasm recognition for spoken dialogue systems. In *Proceedings of the INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing*.

Oren Tsur, Dmitry Davidov, and Ari Rappoport. 2010. ICWSM - A Great Catchy Name: Semi-Supervised Recognition of Sarcastic Sentences in Online Product Reviews. In *Proceedings of the Fourth International Conference on Weblogs and Social Media (ICWSM-2010)*, ICWSM 2010.

Michael Wiegand, Josef Ruppenhofer, and Dietrich Klakow. 2013. Predicative adjectives: An unsupervised criterion to extract subjective adjectives. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 534–539, Atlanta, Georgia, June. Association for Computational Linguistics.

T. Wilson, P. Hoffmann, S. Somasundaran, J. Kessler, J. Wiebe, Y. Choi, C. Cardie, E. Riloff, and S. Patwardhan. 2005a. OpinionFinder: A System for Subjectivity Analysis. In *Proceedings of HLT/EMNLP 2005 Interactive Demonstrations*, pages 34–35, Vancouver, Canada, October.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005b. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the 2005 Human Language Technology Conference / Conference on Empirical Methods in Natural Language Processing*.