SARCASM DETECTION IN TWITTER

Yaaay! it's a holiday weekend and I have a sarcasm detection project to complete! couldn't be more thrilled!

Erol Özkan
Department of Computer Engineering, Hacettepe University
erolozkan@outlook.com

Abstract - Sentiment Analysis is a technique to identify people's opinion, attitude, sentiment, and emotion towards any specific target such as individuals, events, topics etc. And the sarcasm is a special kind of sentiment that comprise of words which mean the opposite of what you really want to say. It is largely used in social networks like facebook and twitter where people mock or criticize in a way that makes it difficult even for humans to tell if what is said is what is meant. Therefore in order to improve automatic sentiment analysis of data collected from social networks, recognizing sarcastic statements is a mandatory. In this paper, we try to characterize a sarcasm recognizer to identify tweets that is in the most common form of sarcasm which 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 present an approach that automatically learns lists of positive sentiment phrases and negative situation phrases from sarcastic tweets. Then we try to classify new tweets using this kind of contrasting contexts. We show that identifying contrasting contexts using the learned phrases yields promising results for sarcasm recognition task.

Keywords—Natural language processing, Sarcasm Detection, Irony Detection, Satire Detection, Tweets, Twitter, Opinion mining, Parsing, POS tagging, Sentiment, #Sarcasm, #Irony, #Satire ,#ToSumUp, #HardWork.

I. INTRODUCTION

Twitter has become one of the biggest web destinations for users to express their opinions and thoughts. Throughout the last decade, Twitter content has been increased dramatically. Today, millions of tweets are sent everyday by more than half a million active users. This yielded many companies interested in these data for the purpose of studying the opinion of people towards political events, products and movies. However, due to the informal language used in Twitter and the limitation in terms of characters(only 140 chars), understanding the opinions of users and performing such analysis is quite difficult. Furthermore, presence of sarcasm makes the task even more challenging given that 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. Oxford dictionary defines sarcasm as "the use of irony to make or convey contempt". 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 have a sarcasm detection project to complete! couldn't be more thrilled! #sarcasm." In this case, the hashtag #sarcasm reveals the intended sarcasm, however we don't always have the benefit of an explicit sarcasm label.

Sarcasm appear in different forms and patterns. And these patterns are really difficult to recognize for computers. However it is observed that, in the realm of Twitter, the very common structure in sarcastic tweets is that they have positive/negative contrast between a sentiment and a situation. In other words, sarcastic tweets often express a positive sentiment in reference to a negative activity or state. Some examples matches for this schema given in TABLE I.

TABLE I
SOME TWEET EXAMPLES THAT CONTAIN SARCAMS

Oh how I love staying at home at weekends doing NLP project every week.
Thoroughly enjoyed shoveling the driveway today!
I absolutely adore it when I get another A3 instead of an A1. ☹
I'm so pleased mom woke me up with vacuuming my room this morning.

The sarcasm in these tweets arises from the existence 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 purpose of this research is to identify the sarcasm that originate from this kind of schema which is including a positive sentiment referring to a negative situation.

Initially, in order to identify sarcasm; we must learn to recognize phrases that correspond to negative situations and positive sentiment. To do this; First, we extract unenjoyable activities which include going to the dentist, taking an exam, having to work on holidays and undesirable states include being ignored, having no friends, and feeling sick. We use tweets that contain sarcasm as positive instances for the learning process. The algorithm begins with a single seed word, "love", and a large set of sarcastic tweets. It then learns negative situation phrases that follow a positive sentiment (the seed word "love"). Secondly, we learn positive sentiment phrases that occur near a negative situation phrase. 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.

The remainder of this paper is structured as follows. Next section describes some of the related work along with what kind of method they used to detect sarcasm. Section III step by step describes in details our proposed method. Experimental results, concluding results, remarks and future scope are presented in Section 4 and Section 5.

II. Related Work

Sarcasm detection can be classified into three categories, namely lexical, pragmatics and hyperbole.

Lexical feature includes text properties such as uni-gram, bi-gram, tri-gram and n-gram. Kreuz[1] introduced the lexical feature, that play a vital role to recognize irony and sarcasm in text. Utsumi[2] suggested that extreme adjectives and adverbs often provide an implicit way to display negative attitudes. Kreuz[1] in his subsequent work along with Caucci[3] used these lexical and syntactic features to recognize sarcasm in text and also discussed the role of different lexical factors, such as an interjection and punctuation symbols. A semi-supervised approach was introduced by Davidov[4] to detect sarcasm in tweets and Amazon product reviews. They used two interesting lexical features, namely pattern-based (high frequency words and content words) and punctuation-based to build a weighted k-nearest neighbor classification model to perform sarcasm detection.

Pragmatic features include symbolic or figurative texts like smilies, emoticons, replies. Kreuz[1] introduced the concept of pragmatic features for the first time. This feature is very much helpful to identify sarcasm in textual data. GonzlezIbnez[5] explored it further with emoticons, smiles and replies. They developed a system using the pragmatics features to identify sarcasm in tweets. Tayal[6] used pragmatic features to identify sarcasm in political tweets. Rajadesingan[7] used a systematic approach for sarcasm detection in tweets and used psychological and behavioral pragmatic features of an user with their present and past tweets.

Hyperbole plays the most important role in order to identify sarcasm in the text. A combination of the properties such as intensifier, interjection, quotes, punctuations etc. in the text is called hyperbole. Lunando[8], focused only on interjection words such as aha, bah, nah, wah, wow, yay, uh, etc. for sarcasm identification. They conclude that, if the text contains interjection words, it has more tendency to be classified as sarcastic. Liebrecht[9] focused on hyperbole to detect sarcasm in tweets as utterance with a hyperbole ('fantastic weather' when it rains) is identified as sarcastic with more ease than the utterance without a hyperbole ('the weather is good' when it rains). The utterance with the hyperbolic 'fantastic' may be easier to interpret more sarcastic than the utterance with the non-hyperbolic 'good'. Filatova[10] focused on hyperbole features to identify sarcasm in document level text as only a phrase or a sentence is not sufficient for sarcasm detection. They considered the context of a sentence and the surrounding sentences to improve the accuracy of the detection.

There are five types of sarcasm occurring in the text. They are (i)when text sentiment conflict with text situation. (ii)when text contradicts a fact. (iii)when the authors likes and dislikes conflict with the text. (iv)when a text contains sarcasm hashtag at the end. (v)when a text conflict with the fact about any event such as a festival, birthday, sports, etc. A summary of the types is shown in TABLE II.

TABLE II
Types Of Sarcasm Occur In Text[11]

T1	Contrast between positive sentiment and negative situation
T2	Contrast between negative sentiment and positive situation
T3	Fact negation - Text contradicting a fact
T4	Likes and dislikes prediction (behaviour based)
T5	Lexical analysis(sarcasm hashtag based)
T6	Temporal knowledge extraction (tweets contradicting facts about event)

TABLE III
Types Of Feature[11]

F1	Lexical- unigram, bi-gram, tri-gram, n-gram, #hashtag
F2	Pragmatic- smiles, emoticons, replies
F3	Interjection- yay, oh, wow, yeah, nah, aha, etc.
F4	Intensifier- adverb, adjectives
F5	Punctuation mark- !!!!!, ????
F6	Quotes- " " , ' '
F7	Pattern based- high freq. words and content words

TABLE IV
Types Of Domain[11]

D1	Tweets of Twitter
D2	Online product reviews
D3	website comments
D4	Google Books
D5	Online Discussion Forums

Features play an important role in the detection of sarcasm in the text. In the literature, researchers used different features. A list of these features is given in TABLE III. These are (i)Hashtag, if any text contains hashtag sarcasm, then its easy to identify the text as sarcastic. (ii)There are some lexical feature as uni-gram, bi-gram, tri-gram and n-gram through which one can easily identify sarcasm in text. Ex: bi-gram "amazing night" people often say this phrase sarcastically. (iii)Pragmatic feature like emoticons, smiles, replies, etc. These features often used by the author to write sarcastic text. (iv)Another feature which widely used in sarcastic text is an intensifier. Adverbs and adjectives are often used as intensifiers such as fantastic weather, so pleased, etc. (v)There are some interjection words like wow, oh, uh etc. which are used highly in sarcastic texts. Similarly, punctuation mark, quotes, etc. play big role in identification of sarcasm. (vi)There are some high frequency words which also play major role in recognition of sarcasm in text.

Ongoing research on sarcasm detection, researchers used data in various domains. A list of these domains is given in TABLE IV. They are tweets, product reviews, online discussion forum, google books, and website comments.

A consolidated table showing features available, type of sarcasms and domains that are deployed are shown in TABLE V.

TABLE V
PREVIOUS STUDIES IN SARCASM DETECTION[11]

Study	Types of Sarcasm						Types of feature							Domain				
	T1	T2	T3	T4	T5	T6	F1	F2	F3	F4	F5	F6	F7	D1	D2	D3	D4	D5
OZKAN, 2016	√						√							√				
Kreuz et al., 1995					√		√	√		√				√				
Utsumi et al., 2000					√					√				√				
Kreuz et al., 2007					√		√		√		√			√				
Tsur et al., 2010	√			√	√		√				√		√		√			
Davidov et al., 2010					√		√				√			√	√			
Gonzlez-Ibnez et al., 2011					√		√	√						√				
Filatova et al., 2012			√	√				√	√		√	√			√			
Riloff et al., 2013	√				√		√							√				
Lunando et al., 2013					√				√					√				
Liebrecht et al., 2013	√				√		√			√	√			√				
Rajadesingan et al., 2014	√			√	√		√	√						√				
Tungthamthiti et al., 2014	√				√		√			√				√				
Peng et al., 2014	√				√		√						√	√				
Raquel et al., 2014					√			√					√	√	√	√		√
Kunneman et al., 2014					√		√		√	√	√			√				
Barbieri et al. 2014					√		√							√				
Tayal et al., 2014					√		√	√				√		√				
Nitin et al., 2014			√	√	√	√	√							√				
Pielage et al., 2014					√		√	√						√	√	√	√	√

III. METHOD

Sarcasm, in its simplest definition, is that "saying the opposite of what you mean". Therefore, to detect sarcasm in other words to create a sarcasm classifier for tweets; we need to recognizes contexts that contain a positive sentiment contrasted with a negative situation.

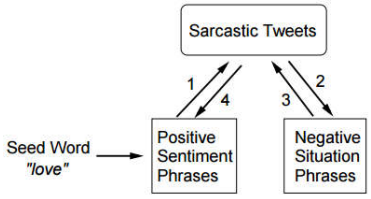

Fig. 1. Positive Sentiment and Negative Situation Phrases

To do that, our approach first learns negative situations using only the seed word "love" from a collection of sarcastic and not sarcastic tweets as inputs, then it explores rich phrasal lexicons of positive sentiments. Finally, it labels the new tweets based on whether they contain a contrasting ideas or not. A key factor that makes the algorithm work is the assumption that if you find a positive sentiment followed by a negative situation in a sarcastic tweet like shown in "Fig. 1", then you have found the source of the sarcasm.

A. The Data

For the learning process, we first used 12,149 sarcastic tweet set to induce positive sentiment and negative situation phrase candidates. Then used 174,391 tweet set that is just random tweets to calculate probabilities values for those candidates.

We assumed the tweets that contain a sarcasm hashtag as positive instances, and considered random data to be negative instances of sarcasm. We take no account of nor the positive instances which is not sarcastic in nature but it just contains the hashtag #sarcasm nor the negative instances that is sarcastic in nature but it does not contain the hashtag #sarcasm. It is expected that a very small percentage of these tweets will be noisy.

B. Learning Process of Negative Situations Phrases

For learning process we assume that many sarcastic tweets contain the below structure:

[+ VERB PHRASE] [– SITUATION PHRASE]

Process begins with a single seed word, "love", which seems to be the most common positive sentiment verb in sarcastic tweets. Given all tweets we first extract the tweets that contains the word "love". Our assumption here infers that "love" is probably followed by an expression that refers to a negative situation which both together forms the source of sarcasm.

After extracting the tweets that contain the word "love", we apply CMU's part-of-speech tagger designed for tweets to this data set. The purpose here to filter the candidate list based on POS patterns to keep only n-grams that have a desired syntactic structure. It is desired to learn possible verb phrase 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. These uni-gram and bi-gram patterns are listed in TABLE VI.

TABLE VI
POS PATTERNS

V – uni-gram
V+V – bi-gram
V+ADV - bi-gram
ADV+V - bi-gram
"TO"+V - bi-gram
V+NOUN - bi-gram
V+PRO - bi-gram
V+ADJ. - bi-gram

Here, for example, for uni-grams there is just a single V tag that covers all types of verbs. For bi-grams; 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". The remaining bigram patterns capture verb phrases that include a verb and adverb, infinitive form (e.g., "to clean"), a verb and noun phrase (e.g., "shoveling snow"), or a verb and adjective (e.g., "being alone). POS-based trigram patterns excluded in order to not to compromise efficiency.

After eliminating the candidates that contains the word "love" and then the ones that does not follow our desired syntactic structure, we extract every 1- gram, 2-gram, and 3-gram that occurs immediately after the word "love" as negative situation candidates for all tweets.

As an example, consider the tweet in Figure 2, where "love" is the positive sentiment:

I love staying at home at weekends doing NLP project.

It delivers our both conditions which are including the word love and satisfying the desired syntactic structure. So given this sentence, we extract three n-grams as candidate negative situation phrases:

Staying , staying at, staying at home

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:

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

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.

Finally, we rank the candidate phrases based on this probability, we select the top 30 phrases and add them to the negative situation phrase list.

C. *Learning Process of Positive Sentiment Phrases*

For the purpose of learning positive sentiment phrases, we do same operations in reverse order. We collect phrases that potentially convey a positive sentiment by extracting n-grams that precede a negative situation phrase in a sarcastic tweets. To learn positive sentiment verb phrases, we extract every 1-gram and 2-gram that occurs immediately before a negative situation phrase.

For the process; first, we eliminate them tweets that does not contain the negative situation phrase we want. Then we apply the POS tagger and filter out the tweets that does not satisfy our desired syntactic structure. But, in here, differently we only retain n-grams that contain at least one verb and consist only of verbs and (optionally) adverbs preceding the negative situation phrase. Finally, after extracting 1 and 2 grams that satisfy our desired syntactic structure, 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:

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

IV. EXPERIMENTATION RESULTS

The learning process learns positive sentiments and negative situations. In our experiments, 30 positive sentiment verb phrases with 135 negative situation phrases have been learned. TABLE VII shows 30 positive verb phrases, and some of the negative situation phrases learned by our proposed method. Here shorter phrases corresponds to more general concepts than the longer phrases therefore they match more contexts in general.

TABLE VII
EXAMPLES FROM LEARNED PHRASES

Positive Verb Phrases (30)	love, enjoy, excited, asked, get, wait, appreciate, stops, keeps, got, missed, start, needs, keep, stop, pumped, decided, wants, wanted, stopped, used to, didn't want, decides not, loves, go, want, need, wake up, gonna be
Negative Situation Phrases (135)	waiting, staying, falling, cuddling, shoveling, living, missing, picking, wearing, studying, working, babysitting, bein, losing, arguing, gettin, buying, standing, looking, texting, walking, using, making, wake, when people think, when people try, when people take, when people call, when doesn't text, when people assume, to be ignored, when you text, when this happens, when people spread, people who think, driving, learning, sleeping, knowing, to go, when people say, to hear, to see, when people read, to get, to know, when it rains, called, call, cancelled, married, invited, replying, posting, paid, stuck, tweeting, telling, run, to talk, to read, to play, to sit, to eat, to stay, back to sleep…

After exploring positive sentiment phrases and negative situation phrases, we tried to determine how well our system performs. Therefore, to calculate what percentage we classified correctly, we first calculate precision and recall. Then use these values to calculate F1 score which can be interpreted as a weighted average of the precision and recall.

The relative contribution of precision and recall to the F1 score are equal. The formulas for precision, recall and F1 score are shown in TABLE VIII respectively.

TABLE VIII
PRECISION, RECALL AND F1 SCORE

Precision	TP/(TP+FP)
Recall	TP/(TP+FN)
F1 Score	2 * (Precision * Recall) / (Precision + Recall)

Then we test our results on a different tweet collection from we used for training. The test set contains total 3000 tweets which is 693 of them are sarcastic tweets. We classify test set tweets using 4 different approach. We calculate precision, recall and f-score for all them. Test results are show in TABLE IX.

TABLE IX
TEST RESULTS

	Precision	Recall	F1 Score
Only Positive Sentiment	0.33 (357/1070)	0.51 (357/693)	0.40
Only Negative Situation	0.36 (295/815)	0.42 (295/693)	0.39
PS and NS With No Order	0.39 (203/516)	0.29 (203/693)	0.33
PS followed by NS	0.46 (103/221)	0.14 (103/693)	0.22

Our approach manages to attain a maximum 0.46 precision, a maximum 0.42 recall and a maximum 0.40 F1 Score respectively in tweets with sarcastic hashtag. In addition, picking longer negative situation and positive sentiment phrases increase precision, however it decrease recall since they cover specific examples. On the other hand, picking shorter phrases increase recall while decreasing recall since they cover more general examples.

V. CONCLUSION AND FUTURE SCOPE

Sarcasm is indeed a complex and rich linguistic phenomenon. Even humans have difficulties to understand it time to time. Furthermore, it requires the world knowledge. For instance, when someone says "Yaaay! it's a holiday weekend and I have a sarcasm detection project to complete! couldn't be more thrilled!", he may be telling the truth and really intends to say that he is enjoying doing NLP project or he means just the opposite that he doesn't not like doing NLP project at weekends at all which makes the sentence very sarcastic.

To sum up, our work is concentrated on only one type of sarcasm which is actually very common in tweets: contrast between a positive sentiment and negative situation. The phrases that we learned were limited to specific syntactic structures and we required the contrasting phrases to appear in a highly constrained context. For the purpose of identifying this contrasting context, we presented a method that learns first negative activities from the seed word "love" then the positive sentiment phrases from learned negative activities and situations. We showed that these lists of phrases really can be used to classify tweets according to whether they contain sarcasm or not. In future, efficiency and accuracy of detecting sarcasm can further be increased by using more features along with this method which is identifying and using positive/negative contrast

REFERENCES

[1] R. J. Kreuz and R. M. Roberts, "Two cues for verbal irony: Hyperbole and the ironic tone of voice," Metaphor and symbol, vol. 10, no. 1, pp. 21–31, 1995.

[2] A. Utsumi, "Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony," Journal of Pragmatics, vol. 32, no. 12, pp. 1777–1806, 2000.

[3] R. J. Kreuz and G. M. Caucci, "Lexical influences on the perception of sarcasm," in Proceedings of the Workshop on computational approaches to Figurative Language, ACL, pp. 1–4, 2007.

[4] O. Tsur, D. Davidov, and A. Rappoport, "Icwsm-a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews.," in ICWSM, pp. 162–169, 2010.

[5] R. Gonzalez-Ib ´ anez, S. Muresan, and N. Wacholder, "Identifying sar- ´ casm in twitter: a closer look," in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-vol 2, pp. 581–586, 2011.

[6] D. Tayal, S. Yadav, K. Gupta, B. Rajput, and K. Kumari, "Polarity detection of sarcastic political tweets," in Computing for Sustainable Global Development (INDIACom), 2014 International Conference on IEEE, pp. 625–628, 2014.

[7] A. Rajadesingan, R. Zafarani, and H. Liu, "Sarcasm detection on twitter: A behavioral modeling approach," in Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, pp. 97–106, 2015.

[8] E. Lunando and A. Purwarianti, "Indonesian social media sentiment analysis with sarcasm detection," in Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on IEEE, pp. 195–198, 2013.

[9] C. Liebrecht, F. Kunneman, and A. van den Bosch, "The perfect solution for detecting sarcasm in tweets# not," Association for Computational Linguistics, pp. 29–37, 2013.

[10] E. Filatova, "Irony and sarcasm: Corpus generation and analysis using crowdsourcing.," in LREC, pp. 392–398, 2012.

[11] K. Bharti, S. Babu, K. Jena, "Parsing-based Sarcasm Sentiment Recognition in Twitter Data," in IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 2015.