This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2016.2594194, IEEE Access

A Pattern-Based Approach for Sarcasm Detection on Twitter

Mondher Bouazizi, Tomoaki Ohtsuki

Abstract—Sarcasm is a sophisticated form of irony widely used in social networks and microblogging websites. It is usually used to convey implicit information within the message a person transmits. Sarcasm might be used for different purposes such as criticism or mockery. However, it is hard even for humans to recognize. Therefore, recognizing sarcastic statements can be very useful to improve automatic sentiment analysis of data collected from microblogging websites or social networks. Sentiment analysis refers to the identification and aggregation of attitudes and opinions expressed by Internet users towards a specific topic.

In this paper we propose a pattern-based approach to detect sarcasm on Twitter. We propose four sets of features that cover the different types of sarcasm we defined. We use those to classify tweets as sarcastic and non-sarcastic. Our proposed approach reaches an accuracy of 83.1% with a precision equal to 91.1%. We also study the importance of each of the proposed sets of features and evaluate its added value to the classification. In particular we emphasize the importance of pattern-based features for the detection of sarcastic statements.

Index Terms—Twitter, Sentiment Analysis, Sarcasm Detection, Machine Learning.

1 Introduction

Twitter became one of the biggest web destinations for people to express their opinions, share their thoughts and report real-time events, etc. Throughout the previous years, Twitter content continued to increase, thus constituting a typical example of the so-called big data. Today, according to its official website[1], Twitter has more than 288 million active users, and more than 500 million tweets are sent every day. Many companies and organizations have been interested in these data for the purpose of studying the opinion of people towards political events [1], popular products [2] or movies [3].

However, due to the informal language used in Twitter and the limitation in terms of characters (i.e., 140 characters per tweet), understanding the opinions of users and performing such analysis is quite difficult. Furthermore, presence of sarcasm makes the task even more challenging: sarcasm is when a person says something different from what he means. Liebrecht et al. [10] discussed how sarcasm can be a polarity-switcher, and Maynard et al. [11] proposed a set of rules to decide on the polarity of the tweet (i.e., whether it is positive or negative) when sarcasm is detected.

The online Oxford dictionary[2] defines sarcasm as *"the use of irony to make or convey contempt"*. Collins dictionary[3] defines it as *"mocking, contemptuous, or ironic language intended to convey scorn or insult"*. However, sarcasm is a deeper concept, highly related to the language, and to the common knowledge.

Although different from one another, sarcasm and irony have been studied as two close and very correlated concepts [4] [5] [6] or even as the same one [7] [8] [9]. The Free Dictionary[4] defines it also as a form of irony that is intended to express contempt. Since most of the focus on sarcasm is to enhance and refine the existing automatic sentiment analysis systems, we also use the two terms synonymously.

Some people are more sarcastic than others, however, in general, sarcasm is very common, though, difficult to recognize. In general, people employ sarcasm in their daily life not only to make jokes and be humorous but also to criticize or make remarks about ideas, persons or events. Therefore, it tends to be widely used in social networks, in particular microblogging websites such as Twitter. That being the case, the state of the art approaches of sentiment analysis and opinion mining tend to have lower performances when analyzing data collected from such websites. Maynard et al. [11] show that sentiment analysis performance might be highly enhanced when sarcasm within the sarcastic statements is identified. Therefore, the need for an efficient way to detect sarcasm arises.

In this paper, we propose an efficient way to detect sarcastic tweet. Although it does not need an already-built user knowledge base as in the work of Rajadesingan et al. [12], our approach considers the different types of sarcasm and detect the sarcastic tweets regardless of their owners or their temporal context, witch a precision that reaches 91.1%.

Therefore, the main contributions of this paper are as follows:

1) We identify the main purposes for which sarcasm is used in social networks.
2) We propose an efficient way to detect sarcastic tweets, and study how to use this information (i.e., whether the tweet is sarcastic or not) to enhance the accuracy of sentiment analysis.

Mondher Bouazizi and Tomoaki Ohtsuki are with the Graduate School of Science and Technology, Keio University, 3-14-1 Hiyoshi, Kouhoku-ku, Yokohama 223-8522, Japan. E-mail: bouazizi@ohtsuki.ics.keio.ac.jp (Mondher Bouazizi), ohtsuki@ics.keio.ac.jp (Tomoaki Ohtsuki)

[1] http://about.twitter.com/company/

[2] http://www.oxforddictionaries.com/

[3] http://www.collinsdictionary.com/

[4] https://www.thefreedictionary.com

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2016.2594194, IEEE Access

3) We study the added value of the different sets of features used, in particular, in terms of precision of detection.

The remainder of this paper is structured as follows: Section 2 presents our motivation for this work and Section 3 describes some state of the art work related to our proposed approach. Section 4 describes our proposed approach for sarcasm detection. In section 5 we present and discuss the obtained results of the approach, and Section 6 concludes this work.

2 Motivations

As mentioned above, the identification of sarcasm helps enhancing sentiment analysis task when performed on microblogging websites such as Twitter. Sentiment analysis and opinion mining rely on emotional words in a text to detect its polarity (i.e., whether it deals *"positively"* or *"negatively"* with its theme). However, the appearance of the text might be misleading. A typical example of that is when the text is sarcastic. In Twitter, such sarcastic texts are very common. *"All your products are incredibly amazing!!!"* might be considered as a compliment. However, considering the following tweet *"Did I say incredibly?? Well, it's true, nobody would believe that. They break the second day you buy them -_-"*, the user explicitly explains that he did not mean what he said. Although some users indicate they are being sarcastic, most of them do not. Therefore, it might be indispensable to find a way to automatically detect any sarcastic messages.

Through their work, Rajadesingan et al. [12] highlighted the limitations of some state of the art tools that perform sentiment analysis, when more sophisticated forms of speech such as sarcasm are present. They explained why sarcasm is hard to detect even by humans, and showed how the nature of tweets makes it even more complicated. Therefore arise the importance of detection of sarcastic utterances in Twitter.

However, several challenges arise and make the task complicated. Joshi et al. [13] highlighted 3 main challenges which are i) the identification of common knowledge, ii) the intent to redicule, and iii) the speaker-listener (or reader in the case of written text) context.

On a related context, even though Brown et al. [4] stated that sarcasm *"is not a discrete logical or linguistic phenomenon"*, works such as [8] [9] were proposed to identify sarcastic writing patterns to decide on whether or not an utterance is sarcastic. During our experiments as well as while manually annotating tweets, we noticed that such patterns exist, in particular among non-native speakers of English. Therefore, we focus on detecting and collecting such patterns from a manually annotated dataset, and we quantify them so that we can judge whether or not a given tweet is sarcastic by comparing patterns extracted from it to them.

Throughout this work, we present a pattern-based framework that performs the task of sarcasm detection, a framework relatively easy to implement, and that presents performances competitive to those of more complex ones.

3 Related Work

In the last few years, more attention has been given to Twitter sentiment analysis by researchers, and a number of recent papers have been addressed to the classification of tweets. However, the nature of the classification and the features used vary depending on the aim. Sriram et al. [14] used non-context-related features such as the presence of slangs, time-event phrases, opinioned words, and the Twitter user information to classify tweets into a predefined set of generic classes including events, opinions, deals, and private messages. Akcora et al. [15] proposed a method to identify the emotional pattern and the word pattern in Twitter data to determine the changes in public opinion over the time. They implemented a dynamic scoring function based on Jaccard's similarity [16] of two successive intervals of words and used it to identify the news that led to breakpoints in public opinion.

However, most of the works focused on the content of tweets and were conducted to classify tweets based on the sentiment polarity of the users towards specific topics. A variety of features was proposed. Not only they include the frequency and presence of unigrams, bigrams, adjectives, etc. [17], but they also include non-textual features such as emoticons [18] (i.e., facial expressions such as smile or frown that are formed by typing a sequence of keyboard symbols, and that are usually used to convey the writer's sentiment, emotion or intended tone) and slangs [19]. Dong et al. [20] proposed a target-dependent classification framework which learns to propagate the sentiments of words towards the target depending on context and syntactic structure.

Sarcasm, on the other hand, and irony in general have been used by people in their daily conversations for a long time. Therefore, sarcasm has been subject to deep studies form psychological [21] and even neurobiological [22] perspectives. Nevertheless, it has been studied as a linguistic behavior characterizing the human being [12]. In this context, researchers have recently been interested in sarcasm, trying to find ways to automatically detect it when it is present in a statement. Although some studies such as [4] highlighted that, unlike irony, sarcasm *"is not a discrete logical or linguistic phenomenon"*, many works have been proposed and present high accuracy and precision.

Burfoot et al. [23] introduced the task of filtering satirical news articles from true newswire documents. They introduced a set of features including the use of profanity and slangs and what they qualified of *"semantic validity"*; and used SVM classifier to recognize satire articles.

Campbell et al. [24] studied the contextual components utilized to convey sarcastic verbal irony and proposed that sarcasm requires the presence of four entities: allusion to failed expectation, pragmatic insincerity, negative tension and presence of a victim, as well as stylistic components.

Nevertheless, other works have been proposed to represent sarcasm. Some of these representations are given in [13] as follows:

- Wilson et al. [25] suggested that sarcasm arises when there is a situational disparity between the text and the context.
- Ivanko et al. [26] suggested that sarcasm requires a 6-tuple consisting of a speaker, a listener, a context, an utterance, a literal proposition and intended proposition.

- Giora et al. [27] suggested that sarcasm is a form of negation in which an explicit negation marker is lacking. This implies that the sarcasm is namely a polarity-shifter.

As for the task of detection itself, several goals were defined. Tepperman et al. [28] studied the occurrence of the expression *"yeah right!"*, and whether it appears in a sarcastic context or not. They proposed an approach to automatically detect sarcasm present in spoken dialogues, using prosodic, spectral and contextual cues. However, this represents the main shortcoming for their approach: absence of such components makes it impossible to detect sarcasm. In other words, although the approach itself is very effective in detecting when a specific expression is sarcastic, this approach is unable to detect any type of sarcasm that might occur. Veale et al. [29] annotated the occurrences of similes such as *"as cool as a cucumber"* into ironic or not. This works presents the same shortcoming as that of Tepperman et al. [28]. Barbieri et al. [30] proposed to classify texts into politics, humor, irony and sarcasm. Ghosh et al. [31] formulated the task of sarcasm detection as a sense disambiguation task where a word can have a literal sense or a sarcastic one, and therefore, through detecting the sense of the word, sarcasm can be detected. Wang et al. [32] suggested that, rather than trying to detect whether a tweet is sarcastic or not, it makes more sense to take into account the context: they modeled the problem as a sequential classification task. However, most of the works simply aim to classify a set of texts as sarcastic and non-sarcastic.

Davidov et al. [9] and Tsur et al. [8] proposed a semi-supervised sarcasm identification algorithm. They experimented on two data sets: one from amazon and the other from Twitter. The results they obtained were interesting, though their approach relies on the frequency of appearance of words which might be misleading if the training set is not balanced in terms of topics it deals with or if the data are not big enough. In addition, it treats what is called *"Context Words"* in the same way regardless of their grammatical function. It also does not make difference between sentimental words and non sentimental words. Patterns that do not consider the emotional content of words, or discard some emotional words because of their low presence might reduce the potential of the approach.

Maynard et al. [11] relied on hashtags that Twitter users employ in their tweets to identify sarcasm in Twitter. They also studied how the detection of sarcasm can highly enhance the sentiment analysis of tweets, and proposed a rule to decide on the polarity of the tweet (i.e., whether it is positive or negative) depending on the apparent sentiment of the tweet and the content of the hashtag.

Riloff et al. [33] proposed a method to detect a specific type of sarcasm, where a positive sentiment contrasts with a negative situation. They introduced a bootstrapping algorithm that uses the single seed word *"love"* and a collection of sarcastic tweets to automatically detect and learn expressions showing positive sentiment and phrases citing negative situations. Their approach shows some potentials. However, most of the sarcastic tweets in Twitter do not fall in the aforementioned category of sarcasm. In addition, the approach relies on the existence of the all possible *"negative situations"* on the training set, which makes it less efficient when dealing with new tweets.

Rajadesingan et al. [12] went deeper and dealt with the psychology behind sarcasm. They introduced a behavioral modeling for detecting sarcasm in Twitter. They identified different forms of sarcasm and their manifestation in Twitter, and demonstrated the importance of historical information collected from the past tweets for sarcasm detection. Although, it has proven to be very efficient, the approach is less performant when there is no previous knowledge about the user. Most of the features extracted rely on data collected from previous tweets to judge. For a realtime stream of tweets, where random users are posting tweets, it is hard to run the approach, the size of the knowledge-base grows very fast, and the training should be redone each time based on the new tweets collected (i.e., since the previous tweet has the highest impact on the current one, the new tweet should be taken into consideration for the next iteration).

Muresan et al. [34] proposed a method to construct a corpus of sarcastic Twitter messages, where the author of the tweet provides the information whether or not a tweet is sarcastic. Throughout their work, they investigated the impact of lexical and pragmatic factors on machine learning performance to identify and detect sarcastic tweets and ranked the features according to their contribution to the classification.

Fersini et al. [35] introduced a Bayesian Model Averaging ensemble that takes into account different classifiers, according to their reliability and their marginal probability predictions to make a voting system more sophisticated than the conventional majority voting one.

Bharti et al. [36] proposed two approaches for detecting sarcastic tweets: the first one is a parsing-based lexicon generation algorithm and the second one uses the occurrences of interjection words.

In general, and based on the method and features used, we can classify these works into 3 categories:

- **Rule-based approaches** such as the work of Maynard et al. [11] and that of Ghosh et al. [31],
- **Semi-supervised approaches** such as the works proposed by Tsur et al. [8], that proposed by Davidov et al. [9] and that proposed by Bharti et al. [36],
- **Supervised approaches** such as the work of Muresan et al. [34], that of Wang et al. [32] and that of Rajadesingan et al. [12].

As for the features used in the supervised approaches they fall mainly into 3 sets:

- n**-gram-based features**, which have been used along with other features in the majority of the works such as the works of Barbieri et al. [30], Riloff et al. [33] and that of Ghosh et al. [31],
- **Sentiment-based features** such as the works of Reyes et al. [37] [38] and Joshi et al. [39],
- **Saracstic pattern-based features** such as the works of Tsur et al. [8], Davidov et al. [9] and Riloff et al. [33], etc.

Other works added the contextual features to enhance the classification, whether the context is the historical context as in [12], the conversation context as in [40] [39] or the topical context as in [32].

In our work, we opt for a supervised approach that learns sarcastic patterns extracted based on the part-of-speech of words used.

4 Proposed Approach

Given a set of tweets, we aim to classify each one of them depending on whether it is sarcastic or not. Therefore, from each tweet, we extract a set of features, refer to a training set and use machine learning algorithms to perform the classification. The features are extracted in a way that makes use of different components of the tweet, and covers different types of sarcasm. The set of tweets on which we run our experiments is checked and annotated manually.

4.1 Data

Throughout the period ranging from December 2014 to March 2015, we collected tweets, using Twitter's streaming API. To collect sarcastic tweets, we queried the API for tweets containing the hashtag *"#sarcasm"*. Although Liebrecht et al. [10] concluded in their work that this hashtag is not the best way to collect sarcastic tweets, other works such as [9] highlighted the fact that this hashtag can be used for this purpose. However, they also concluded that the hashtag cannot be reliable and is used mainly for 3 purposes:

- to serve as a search anchor,
- to clarify the presence of sarcasm in a previous tweet, as in *"I forgot to add #sarcasm so people like you get it!"*,
- to serve as a sarcasm marker in case of a very subtle sarcasm where it is very hard to get the sarcasm without an explicit marker, as in *"Today was fun. The first time since weeks! #Sarcasm"*.

In total, we collected 58 609 tweets with the hashtag *"#sarcasm"*, which we cleaned up by removing the noisy and irrelevant ones, as well as ones where the use of the hashtag does fall into one one of the two first uses of the three described above.

As for non-sarcastic tweets, we collected tweets dealing with different topics and made sure they have some emotional content.

We prepared 3 data sets for our work as follow:

- **Set 1:** this set contains 6000 tweets, half of them are sarcastic, and the other half are not. The tweets on this data set are manually checked and classified depending on their level of sarcasm from 1 (highly non-sarcastic) to 6 (highly sarcastic). The manual annotation is done by two people with no background about the tweets or the users who posted them. They have been asked to attribute the scores. It is important to note that the manual labelling is subject to the annotators' own opinion. Therefore, it is taken into account that the classification is not perfect. However, a sarcastic tweet is never labeled as non-sarcastic, and vice versa. Therefore, this set contains a trustworthy knowledge base that can be used to train our model. Tweets having level of sarcasm equal to 3 are mostly ones that, without the hashtag *"#sarcasm"*, are very close those of level 4 or 5. In other terms, it is very hard for a human, with no background about the tweet, to tell whether it is sarcastic or not. The hashtag *"#sarcasm"* has not been removed yet when the annotation is done. This first set is used to train our model. Therefore, in the rest of this work, it will be referred to as the *"training set"*. The number of sarcasm levels is also referred to as N_S and is equal to 6.
- **Set 2:** this set contains 1128 sarcastic tweets, and 1128 non-sarcastic ones. Sarcastic tweets are collected as described above (i.e., by querying Twitter API). Yet, no manual check is done, which makes it a very noisy data set. However, to reduce the noise, we filtered-out the non-english tweets, very short tweets (i.e., that have less than 3 words), and those which contain URLs. In most of the cases, URLs refer to photo links. We believe that part of the sarcasm is included in the photo, therefore we discard them. This data set is used during our experimenting process to optimize the parameters we defined for our features. In the rest of this work, we will refer to this set as the *"optimization set"*.
- **Set 3:** this set contains 500 sarcastic tweets, and 500 non-sarcastic ones. All tweets are manually checked and classified as sarcastic and non-sarcastic. This set will serve as a test set, and will be used to evaluate the performances of our proposed approach. Therefore, in the rest of this work, it will be referred to as the *"test set"*.

None of the tweets of any of the aforementioned sets is re-used in another. In addition, during our work, we removed the hashtag *"#sarcasm"* from all the tweets.

4.2 Tools

To perform the different Natural Language Processing (NLP) taks (i.e., tokenisation, lemmatization, etc.), we used Apache OpenNLP[5]. However, OpenNLP PoS tagger performs poorly with the given model to tag tweets, due to the irrelevant content and the use of slangs, etc., we used Gate Twitter part-of-speech tagger [41]. This PoS-tagger reaches an accuracy of 90.5% on Twitter data.

To perform the classification, we used the toolkit weka [42] which presents a variety of classifiers. We used libsvm [43] to perform the classification using Support Vector Machine (SVM).

4.3 Features Extraction

Being a sophisticated form of speech, sarcasm is used for different purposes. While annotating the data, the annotators concluded that these purposes fall mostly, but not totally, in three categories: sarcasm as wit, sarcasm as whimper and sarcasm as avoidance.

- **Sarcasm as wit:** when used as a wit, sarcasm is used with the purpose of being funny; the person employs some special forms of speeches, tends to exaggerate, or uses a tone that is different from that when he talks usually to make it easy to recognize.

[5] https://opennlp.apach.org

In social networks, voice tones are converted into special forms of writing: use of capital letter words, exclamation and question marks, as well as some sarcasm-related emoticons.

- **Sarcasm as whimper:** when used as whimper, sarcasm is employed to show how annoyed or angry the person is. Therefore, it tempts to show how bad the situation is by using exaggeration or by employing very positive expressions to describe a negative situation.
- **Sarcasm as evasion:** it refers to the situation when the person wants to avoid giving a clear answer, thus, makes use of sarcasm. In this case, the person employs complicated sentences, uncommon words and some unusual expressions.

Unlike [44], which classifies sarcasm into 4 different types based on how sentiments appear in the text, the observations and classification are done based on why sarcasm is used. Although theses observations are likely to be biased and depend on the annotator's own opinions, we use on these assumptions to build our model. During our work, we rely mainly on writing patterns to detect sarcastic statements; however, other features are extracted and that help to obtain higher classification precision and accuracy. The distinction of purposes highlights the use of some features as we will describe next.

Four families of features are extracted: sentiment-related features, punctuation-related features, syntactic and semantic features, and pattern features.

4.3.1 Sentiment-related Features

A very popular type of sarcasm that is widely used in both regular conversations as well as short messages such as tweets, is when an emotionally positive expression is used in a negative context. A similar way to express sarcasm is to use expressions having contradictory sentiments. This type of sarcasm we qualified as "whimper" is very common in social networks and microblogging websites. Riloff et al. [33] show that this type of sarcasm can be identified and detected when a positive statement, usually a verb or a phrasal verb, is collocated with a negative situation (e.g., *"I love being ignored all the time"*). They built a lexicon-based approach that learns the possible positive expressions and negative situations and used it to detect such contrast in unknown tweets. However, learning all possible negative situations requires a big and rich source and might be infeasible because negative situations are unpredictable.

In our work, we opt for a more straight-forward, yet more general approach. We consider any kind of inconsistency between sentiments of words as well as other components within the tweet. Therefore, to identify and quantify such inconsistency, we extract sentimental components of the tweet and count them. For this purpose, we maintain two lists of words qualified as "positive words" and "negative words". The two lists contain respectively words that have positive emotional content (e.g., *"love"*, *"happy"*, etc.) and negative emotional content (e.g., *"hate"*, *"sad"*, etc.). The two lists of words are created using SentiStrength [6] database. This database contains a list of emotional words, where negative words have scores varying from -1 (almost negative) to -5 (extremely negative) and positive words have score varying from 1 (almost positive) to 5 (extremely positive). Using these two lists, we extract two features we denote respectively pw and nw by counting the number of positive and negative words in the tweet.

TABLE 1: PoS-Tags for Words Considered as Highly Emotional

Part of Speech	Part of Speech Tag
Adjectives	"JJ", "JJR", "JJS"
Adverbs	"RB", "RBR", "RBS"
Verbs	"VB", "VBD", "VBG", "VBN", "VBP", "VBZ"

Adjectives, verbs and adverbs have higher emotional content than nouns [45]; therefore positive and negative words that have the associated PoS-tag, shown in TABLE 1, are counted again and used to create two more features that we denote PW and NW and which represent the number of highly emotional positive words and highly emotional negative words.

We then add three more features by counting the number of positive, negative and sarcastic emoticons. Sarcastic emoticons are emoticons used sometimes with sarcastic or ironical statements (e.g., *":P"*). These emoticons are used sometimes when the person is trying to be funny or to show that he is just making a joke (i.e., when sarcasm is used as wit).

Hashtags also have emotional content. In some cases, they are used to disambiguate the real intention of the Twitter user conveyed in his message. For example, the hashtag employed in the following tweet: *"Thank you very much for being there for me #ihateyou"* tells that the user does not really want to thank the addressee, he was rather blaming him for not being there for him. Therefore, we count also the number of positive and negative hashtags.

In addition to the aforementioned features, we extract features related to the contrast between these sentimental components. We first calculate the ratio of emotional words $\rho(t)$ defined as

$$\rho(t) = \frac{(\delta \cdot PW + pw) - (\delta \cdot NW + nw)}{(\delta \cdot PW + pw) + (\delta \cdot NW + nw)} \qquad (1)$$

where t is the tweet, *pw*, *PW*, *nw* and *NW* denote respectively the number of positive words (other than highly emotional ones), that of highly emotional positive words, that of negative words (other than highly emotional ones) and that of highly emotional words. δ is a weight bigger than 1 given to the highly emotional words. In case the tweet does not contain any emotional word, ρ is set to 0. In the rest of this work, δ is set to 3.

We then define 4 features that represent whether there is a contrast between the different components. By contrast we mean the coexistence of a negative component and a positive one within the same tweet. We check the existence of such contrast between words, between hashtags, between words and hashtags and between words and emoticons and use these information as extra features. The final sentiment-related feature vector has 14 features.

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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2016.2594194, IEEE Access

4.3.2 Punctuation-Related Features

Sentiment-related features are not enough to detect all kinds of sarcasm that might be present. In addition, they do not make use of all the components of the tweet. Therefore, more features are to be extracted. As mentioned before, sarcasm is a sophisticated form of speech: not only it plays with words and meanings, but also it employs behavioral aspects such as low tones [47] [48], facial gestures [49] or exaggeration. These aspects are translated into a certain use of punctuation or repetition of vowels when the message is written. To detect such aspects, we extract a set of features that we qualify as punctuation-related features. For each tweet, we calculate the following values:

- Number of exclamation marks
- Number of question marks
- Number of dots
- Number of all-capital words
- Number of quotes

We also add a sixth feature by checking if any of the words contains a vowel that is repeated more than twice (e.g., *"loooooove"*). If such a word exists, the feature is set to *"true"*, otherwise, it is set to *"false"*.

The *"excessive"* use of exclamation marks or question marks, or the repetition of a vowel, particularly in an emotional word, might reflect a certain tone that the user intends to show, however, this tone is not always sarcastic. We believe that these features can be highly correlated with the number of words in the tweet. Some very short tweets which end with many exclamation marks might show surprise rather than sarcasm. Following two examples of tweets in which the use of exclamation marks has two different use cases:

- *"Thank you @laur3en, it was amazing !!!"*
- *"Thanks for another amazing day with your amazing boyfriend!!!!"*

In the first case, the exclamation marks are used to show sincere feelings of gratitude. However, in the second, the exclamation marks serve as an indication of annoyance; the user has no real intension to thank his friend. Although the use of exclamation is not relevant in itself and might not show whether the user is expressing sarcasm or any other emotion; combined with other features, this feature is expected to add value to the classification. We then define one last feature by counting the number of words in the tweet. In total, 7 punctuation-based features are extracted.

4.3.3 Syntactic and Semantic Features

Along with the punctuation-related features, some common expressions are used usually in a sarcastic context. It is possible to correlate these expressions with the punctuation to decide whether what is said is sarcastic or not. Besides, in other cases, people tend to make complicated sentences or use uncommon words to make it ambiguous to the listener/reader to get a clear answer. This is common when sarcasm is used as "evasion", where the person's purpose is to hide his real feeling or opinion by using sarcasm. Hence, we extract the following features that reflects these aspects:

- Use of uncommon words
- Number of uncommon words
- Existence of common sarcastic expressions
- Number of interjections
- Number of laughing expressions

TABLE 2: Expressions Used to Replace the Words of GFI

PoS-tag	Expression
"CD"	[CARDINAL]
"FW"	[FOREIGNWORD]
"UH"	[INTERJECTION]
"LS"	[LISTMARKER]
"NN", "NNS", "NNP", "NNPS",	[NOUN]
"PRP", "PRP$"	[INTERJECTION]
"MD"	[MODAL]
"PB", "RBR", "RBS"	[ADVERB]
"WDT", "WP", "WP$", "WRB"	[WHDETERMINER]
"SYM"	[SYMBOL]

In particular, the feature "Existence of common sarcastic expression" is extracted in the same way we extract the features qualified as *"pattern-related"* (this will be described in detail in the next subsection). Here we used a noisy set of 3000 tweets having the hashtag *"#sarcasm"* (the set has been discarded later and has not been used neither for training nor for test). We extracted all possible patterns of length varying from 3 to 6, we selected the patterns that appeared more than 10 times. Being few in number, we manually checked the list and removed the irrelevant ones. We obtained a list of 13 main patterns including [*love* PRONOUN *when*] (e.g., *"I love it when I am called at 4 a.m. because my neighbour's kid can't sleep!"*), [PRONOUN *be* ADVERB *funny*] (e.g., *"You are incredibly funny -_-"*), etc.

4.3.4 Pattern-Related Features

The patterns selected in the previous subsection, and qualified of *"common sarcastic expression"* are very common, even in spoken language. However, their number is small, they are not unique and most of the tweets in both our training and test sets do not contain them. That being the case, we dig further and extract another set of features. The idea of our pattern-related features is inspired from the work of Davidov et al. [9]. In his approach, the author classified words into two categories: high-frequency words and content words based on their frequency of appearance in his data set and defined a pattern as an *"ordered sequence of high frequency words and slots for content words"*. This approach, although it has some potential to detect sarcasm, presents many shortcomings as shown in Section 3.

Therefore, we propose more efficient and reliable patterns. We divide words into two classes: a first one referred to as *"CI"* containing words of which the content is important and a second one referred to as *"GFI"* containing the words of which the grammatical function is more important. If a word belongs to the first category, it is lemmatized; otherwise, it is replaced it by a certain expression. The expressions used to replace these words are shown in TABLE 2. The classification into classes is done based on the part of speech tag of the word in the tweet. The list of part-of-speech tags, their meaning and to which category we classify them is given in TABLE 3.

We generate the vector of words for each tweet according to the rule defined. For example, the following

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2016.2594194, IEEE Access

TABLE 3: Part-of-Speech Tag Classes

POS Tag	Description	Class
CC	coordinating conjunction	CI
CD	cardinal number	GFI
DT	determiner	CI
EX	existential there	CI
FW	foreign word	GFI
IN	prep./sub. conjunction	CI
JJ	adjective	CI
JJR	adjective, comparative	CI
JJS	adjective, superlative	CI
LS	list marker	GFI
MD	modal	GFI
NN	noun, singular or mass	GFI
NNS	noun plural	GFI
NNP	proper noun, singular	GFI
NNPS	proper noun, plural	GFI
PDT	predeterminer	CI
POS	possessive ending	CI
PRP	personal pronoun	GFI
PRP$	possessive pronoun	GFI
RB	adverb	CI
RBR	adverb, comparative	CI
RBS	adverb, superlative	CI
RP	particle	CI
SYM	Symbol	GFI
TO	to	CI
UH	interjection	GFI
VB	verb, base form	CI
VBD	verb, past tense	CI
VBG	verb, gerund/present participle	CI
VBN	verb, past participle	CI
VBP	verb, sing. present, non-3d	CI
VBZ	verb, 3rd person sing. present	CI
WDT	wh-determiner	GFI
WP	wh-pronoun	GFI
WP$	possessive wh-pronoun	GFI
WRB	wh-abverb	GFI

TABLE 4: Pattern Features

		Pattern length			
		L_1	L_2	$\cdots$	L_N
	1	F_{11}	F_{12}	$\cdots$	F_{1N}
Sarcasm	2	F_{21}	F_{22}	$\cdots$	F_{2N}
level	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
	6	F_{61}	F_{62}	$\cdots$	F_{6N}

PoS-tagged tweet *"@gilbert:_NN you_PRP are_VBP crazy_JJ ,_, who_WP told_VBD you_PRP I_PRP want_VBP to_TO drink_VB with_IN you_PRP !!!!_."* gives, the following pattern vector [NOUN PRONOUN *be crazy who tell* PRONOUN PRONOUN *want to drink with* PRONOUN.]

We define a pattern as an ordered sequence of words. The patterns are extracted from the training set and are taken such as their length satisfies

$$L_{Min} \leq Length(pattern) \leq L_{Max} \tag{2}$$

where L_{Min} and L_{Max} represent the minimal and maximal allowed length of patterns in $words$ and $Length(pattern)$ is the length of the pattern in $words$. The number of pattern lengths is $N_L = (L_{Max} - L_{Min} + 1)$. Therefore, from the example mentioned above, we can extract the following patterns:

- [NOUN PRONOUN *be crazy*]
- [PRONOUN *be crazy*]
- [*be crazy who tell* PRONOUN PRONOUN *want to*]
- etc.

Only patterns that appear at least N_{occ} times in our training set are kept; the others are discarded. In the rest of this work, N_{occ} is set to 2: the value 1 gives lower accuracy and precision and higher values decrease remarkably the number of patterns, and consequently presents lower accuracy. In addition, a pattern that appears in a sarcastic tweet and in a non-sarcastic tweet is discarded. This step is done to filter out patterns that are not related to sarcasm. After the selection, we divide the resulted patterns into N_F sets, where

$$N_F = N_L \times N_S. \tag{3}$$

We create N_F features, as shown in TABLE 4. Each feature F_{ij} of the table represents the degree of resemblance of the tweet to the patterns of degree of sarcasm i and length j. Therefore, given a tweet t, we calculate the resemblance degree $res(p,t)$ of each pattern in the training set p to the tweet t, defined as:

$$res(p,t) = \begin{cases} 1, & \text{if the tweet vector contains the pattern as it is, in the same order,} \\ \alpha \cdot n/N, & \text{if } n \text{ words out of the } N \text{ words of the pattern appear in the tweet in the correct order,} \\ 0, & \text{if no word of the pattern appears in the tweet.} \end{cases}$$

Given N_{ij} the number of patterns collected from the training set having a sarcasm degree i and a length j, the value of the feature F_{ij} is

$$F_{ij} = \beta_j * \sum_{k=1}^{K} res(p_k, t) \tag{4}$$

where β_j is a weight given to patterns of length L_j (regardless of their level of sarcasm). We give different weights for each length of pattern since longer patterns are more likely to have higher impact. F_{ij} as defined measures the degree of resemblance of a tweet t to patterns of level of sarcasm i and length j. K in our work is set to 5, and represents the K closest patterns among the N_{ij} ones described above.

Extension of the training set patterns: Being relatively small in size (i.e., only 6000 tweets), our training set cannot cover all possible sarcastic patterns. Therefore, we enrich it to obtain more patterns. We collected 18 959 tweets containing the hashtag *"#sarcasm"* and 18 959 tweets that do not. We checked if the tweets having the hashtag *"#sarcasm"* contain any of the sarcastic patterns we already extracted from the training set and that have a length equal to or more than 4. If that is the case, we extract the different patterns from the tweet and add them to the list of patterns of the training set keeping in mind the rule we made for the selection of patterns (i.e., if the pattern exists in a non-sarcastic tweet, it is discarded). Although the added tweets are not as reliable as those of the initial training set, we believe that filtering the tweets that contain at least one pattern that is identical to a reliable one is reliable enough given it already contains the hashtag *"#sarcasm"*. We then did the same to the non sarcastic tweet. Thus, we enriched our data set with more

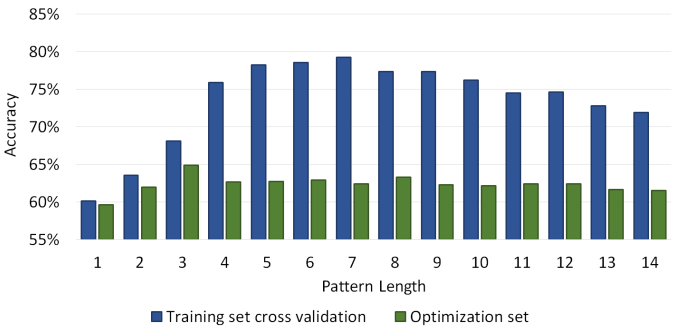

Fig. 1: Accuracy per pattern length for fixed values of $\alpha, \beta_1, \cdots, \beta_{N_L}$

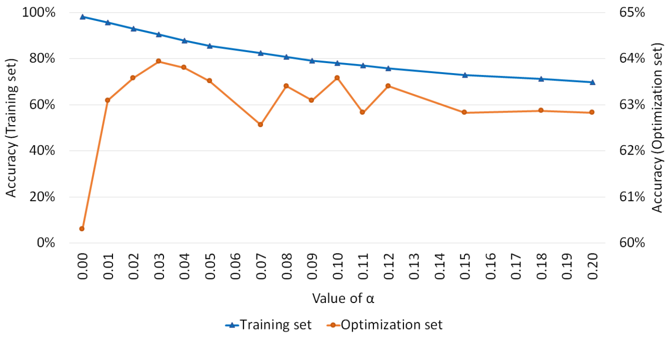

Fig. 2: Accuracy of classification for different values of α

patterns. This step has been done only to get more patterns, therefore, none of the other families of features is concerned by the enrichment.

Pattern-related features as defined give a high flexibility to optimize depending on their contribution. In total we have the following parameters to optimize:

- L_{Min} and L_{Max}
- α
- $\beta_1, \cdots, \beta_{N_L}$

To optimize L_{Min} and L_{Max}, we fixed α and β_i $(i = 1, N_L)$ as follow and tried different values of pattern lengths:

$$\begin{cases} \alpha & = 0.1, \\ \beta_1 = \cdots = \beta_{N_L} & = 1.0. \end{cases}$$

We ran a first simulation on our training set (6000 tweets) and optimization set (2256 tweets), for each pattern length. We obtained the results shown in Fig. 1. The results present the accuracy of the classification of tweets as sarcastic and non-sarcastic. The obtained results show that the patterns having a length are from 4 to 10 give the highest accuracy (i.e., more than 75% accuracy during 10-folds cross validation). Pattern length 3 gives the highest accuracy on our optimization set. Given that the average number of words per tweet is equal to 11.48, we set the parameters L_{Min} and L_{Max} respectively to 3 and 10.

Afterwards, we set Min_{Length} and Max_{Length} as mentioned, kept the values of $\beta_1, \cdots, \beta_{N_L}$ as they are (i.e., equal to 1). We tried different values of α. We ran different simulations on the same data sets using pattern features, for different values of α. Results of the test are given in Fig. 2.

The accuracy of classification varies highly depending on the value of α, that is, the lower the value is, the better the performances are during the cross validation. This is due to the unicity of the patterns. In other terms, in the training set, the patterns derived from each tweet will have the highest score. Thus, the tweet will be classified as the closest to its own patterns. However, in the optimization set, the accuracy is the highest when $\alpha \in \{0.01, 0.1\}$. The optimal accuracy we obtained was for $\alpha = 0.03$ as shown in Fig. 2

Finally, for $\beta_1, \cdots, \beta_{N_L}$, we tried different combinations maintaining the following condition

$$\beta_1 \leq \cdots \leq \beta_{N_L}. \tag{5}$$

The observed results are not very different for all the combinations we tried although we noticed that the closer the values to 1, the better the performances are. The optimal performances we obtained were observed when

$$\beta_n = \frac{n-1}{n+1}. \tag{6}$$

The final values of parameters we set for pattern-related features are as follow:

$$\begin{cases} N_{occ} & = & 2, \\ L_{Min} & = & 3, \\ L_{Max} & = & 10, \\ \alpha & = & 0.03, \\ \beta_n & = & (n-1)/(n+1) \qquad \forall n \in \{3, \ldots, 10\}. \end{cases}$$

In the next section, we evaluate the model we built and present the results of our experiments.

5 Experimental Results

Once the features are extracted, we proceed to our experiments. The Key Performance Indicators (KPIs) used to evaluate the approach are:

- **Accuracy:** it represents the overall correctness of classification. In other words, it measures the fraction of all correctly classified instances over the total number of instances.
- **Precision:** it represents the fraction of retrieved sarcastic tweets that are relevant. In other words, it measures the number of tweets that have successfully been classified as sarcastic over the total number of tweets classified as sarcastic.
- **Recall:** it represents the fraction of relevant sarcastic tweets that are retrieved. In other words, it measures the number of tweets that have successfully been classified as sarcastic over the total number of sarcastic tweets.

We ran the classification using the classifiers *"Random Forest"* [46], "Support Vector Machine" (SVM), "k Nearest Neighbours" (k-NN) and "Maximum Entropy". Table 5 presents the performances of the classifiers on the dataset.

The overall accuracy obtained reaches 83.1% using the classifier Random Forest for an F1-score equal to 81.3%. This accuracy is obtained when setting the parameters of the classifier as follows [46]:

TABLE 5: Accuracy, Precision, Recall and F1-Score of Classification Using Different Classifiers

	Overall Acc.	Precision	Recall	F1-Score
Rand. Forest	83.1%	91.1%	73.4%	81.3%
SVM	60.0%	98.1%	20.4%	33.8%
k-NN	81.5%	88.9%	72.0%	79.6%
Max. Ent.	77.4%	84.6%	67.0%	74.8%

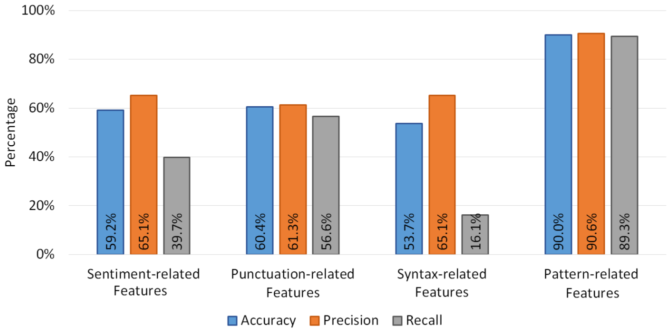

Fig. 3: Accuracy of classification during cross-validation for each family of features

- Number of Features: 20
- Number of Trees: 100
- Seeds: 20
- Max Depth: 0 (unlimited

SVM, on the other hand, presents a precision equal to 98.1% for a low F1-score equal to 33.8%. This means that most of the tweets that were classified as sarcastic are indeed sarcastic. However, a very few percentage of the sarcastic tweets were detected (almost 20%). In other words, SVM is capable of detecting sarcasm with a high precision and the output can indeed be used to refine sentiment analysis, however, it does not cover all the sarcastic tweets. In a real stream of tweets, the number of sarcastic tweets is quite lower than that in the dataset used; therefore, the results obtained mean that only one out of five sarcastic tweets will be detected. Classifiers such as k-NN and Maximum Entropy present a high accuracy and F1-scores, however, the performances of Random Forest are the highest. During the preliminary experiments (i.e., parameters optimization) as well as for the rest of our analysis, the results used are those returned by the classifier Random Forest.

5.1 Performances of Each Set of Features

We first checked the performances of classification of each set of features apart. Figs. 3 and 4 present the performances of the different sets of features.

5.1.1 *During cross-validation*

Fig. 3 shows the performances of classification during cross-validation. We notice that the performances of the pattern-related features is very high during cross-validation. This has been discussed in the previous section: the value of α as chosen makes each tweet in the training set the closest to itself. This explains the very good results obtained by Davidov et al. [9].

TABLE 6: Ratio of Presence of Syntax-Related Features in the Training Set

	True	False	Ratio
Presence of uncommon words	243	5757	4.05%
Presence of common sarcastic patterns	115	5885	1.92%
Presence of interjections	410	5590	6.83%
Presence of laughters	224	5776	3.73%

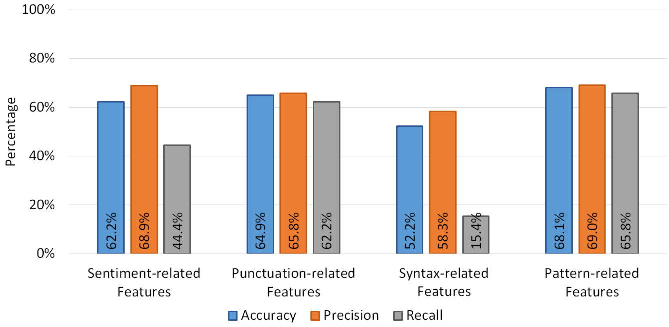

Fig. 4: Accuracy of classification of the test for each family of features

On the other hand, we notice that the syntax-related features present a very low accuracy and recall. The features seem to be not very efficient, if used alone, to classify the tweets as sarcastic and non-sarcastic. One reason is the low presence of these features in the data set. TABLE 6 shows the existence rate of each of the features in the training set. In addition, due to the informal language used in Twitter and the noise it has, the PoS-tagger performances are lower than when applied to a formal text. In particular, the PoS-tagger is not very efficient to detect interjections, it classifies them in many cases as nouns. However, the precision given by this set of features, and which exceeded 65% shows the importance of such features to detect sarcastic components. It refers to the number of sarcastic tweets over the number of tweets judged as sarcastic. Although, they perform poorly, these features might have higher added value when correlated with other features, or if their presence is more frequent.

Punctuation-related features and sentiment-related features have higher prediction rate. They are more efficient, though they perform worse than pattern-related features. They both give an accuracy almost equal to 60%. Furthermore, the precision of sentiment-based features is remarkably higher than the accuracy. In other terms, from the tweets that have been classified as sarcastic, the prediction rate is high. This can be explained by the fact that tweets having contrasting emotional content are likely to be sarcastic. Thus, if detected, they would be classified as sarcastic.

5.1.2 *On a test set*

Fig. 4 shows the performances of classification on our test set. Performance of the classification on unknown data is clearly lower than that during cross-validation. However, we can notice that the sets of features that have the highest merit during cross-validation are the same ones that have the highest merit during the classification of test set tweets.

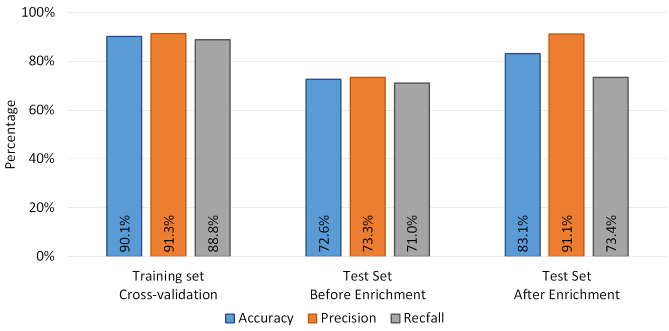

Fig. 5: Accuracy of classification using all features during training set-cross-validation and on the test set

The low accuracy of syntax-related features is due to their low presence in the test set too. As for Pattern-related features, they have higher performances. Accuracy and precision have very close values. This can be explained by the fact that, contrarily to sentiment-based features for example, which check the existence on some characteristics related to sarcasm in the tweets, patterns are extracted from both sarcastic and non-sarcastic tweets, and the closeness to these patterns is checked.

5.2 Overall Performances of the Proposed Approach

Together, the features perform better than each one by itself. Fig. 5 shows the performance of the proposed approach when all the features are used.

During cross validation, both the accuracy and precision are higher than 90%. The recall is lower than 89%. More interestingly, the accuracy obtained for the test set, before enrichment of the patterns, exceeds 72% with a precision higher than 73%. This shows that, if combined, the different sets of features, perform better. Although our data set contains many sarcastic tweets that are hard to identify even by humans (we referred to the hashtag *"#sarcasm"* to classify them), the accuracy obtained is high. The enrichment process added more potential to the approach and increased the accuracy of the classification noticeably. The precision also increased compared to that without enrichment. It reflects the fact that most of the tweets that have been classified as sarcastic really are. Recall, on the other hand, has a lower value, though still better than before enrichment. It shows that, many of the sarcastic tweets were not well classified. As mentioned before, tweets of sarcasm level 3 are very difficult to be distinguished from the non sarcastic ones, therefore, we believe that many of the sarcastic tweets that were not classified as sarcastic fall in this category. Nevertheless, this can be enhanced if we use more tweets for enrichment or in the training set.

To measure the potential of our method, we consider the approach proposed by Riloff et al. [33] as well as the *n*-gram-based approaches as our baseline. In addition to the aforementioned KPIs, we define a fourth one, which is the F˙1 score defined as follow:

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (7)$$

TABLE 7: Performance of the Proposed Approach Compared to the Baseline Ones

	Accuracy	Precision	Recall	F-Score
n-grams	65.9%	82.2%	40.6%	65.9%
Riloff et al. [33]	59.4%	65.0%	40.8%	50.1%
Proposed approach	**83.1%**	**91.1%**	**73.4%**	**81.3%**

It combines the precision and recall, therefore it represents a more reliable KPI to compare different approaches.

The results of the comparison of our approach with the baseline ones are given by TABLE 7. Our proposed approach clearly outperforms the baseline ones, for the used data set: not only it has a higher accuracy and precision, our method's F˙1 score is neatly higher than that of the baseline ones. Although it performs well when detecting a specific type of sarcasm, the approach proposed by Riloff et al. [33], performs poorly in our data set since most of the sarcastic tweets do not fall in the type of sarcasm where a positive sentiment contrasts with a negative situation. This explains the high precision of that approach and its low recall.

Compared to more sophisticated approaches such as that proposed by Davidov et al. [9] or Rajadesingan et al. [12], our approach, although it does not require a big training data set, or a knowledge base of the users, presents competitive results. The two approaches were not reimplemented and run on our data set for the reason that we do not have a previous knowledge of the users as in [12], nor do we dispose of 5.9 million tweets to classify words into context words and highly frequent words as in [9]. However, our proposed presents an F1 score close to that of the approach [9] which is 82.7% (on the Twitter data set) and an accuracy close to that of [12] which is 83.46%.

6 Conclusion

In this work, we proposed a new method to detect sarcasm in Twitter. The proposed method makes use of the different components of the tweet. Our approach makes use of Part-of-Speech-tags to extract patterns characterizing the level of sarcasm of tweets. The approach has shown good results, though might have even better results if we use a bigger training set since the patterns we extracted from the current one might not cover all possible sarcastic patterns.

We also proposed a more efficient way to enrich our set with more sarcastic patterns using an initial training set of 6000 Tweets, and the hashtag *"#sarcasm"*.

In a future work, we will study how to use the output of the current one to enhance the performances of sentiment analysis and opinion mining.

Acknowledgment

The research results have been achieved by "Cognitive Security: A New Approach to Securing Future Large Scale and Distributed Mobile Applications," the Commissioned Research of National Institute of Information and Communications Technology (NICT) , JAPAN.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2016.2594194, IEEE Access

REFERENCES

[1] J. M. Soler, F. Cuartero, and M. Roblizo, "Twitter as a tool for predicting elections results," in *Proc. IEEE/ACM ASONAM*, pp. 1194–1200, Aug. 2012.

[2] S. Homoceanu, M. Loster, C. Lofi, and W-T. Balke, "Will I like it? Providing product overviews based on opinion excerpts," in *Proc. IEEE CEC*, pp. 26–33, Sept. 2011.

[3] U. R. Hodeghatta, "Sentiment analysis of Hollywood movies on Twitter," in *Proc. IEEE/ACM ASONAM*, pp. 1401–1404, Aug. 2013.

[4] R. L. Brown, "The pragmatics of verbal irony," *Language use and the uses of language*, pp. 111–127, 1980.

[5] S. Attardo, "Irony as relevant inappropriateness," Irony in language and thought, pp. 135–174, June 2007.

[6] R. W. Gibbs and J. O?Brien., "Psychological aspects of irony understanding," Journal of Pragmatics, pp. 523–530, Dec. 1991.

[7] H. Grice, "Further notes on logic and conversation," Pragmatics: syntax and semantics, pp. 113–127, Academic Press, 1978.

[8] O. Tsur, D. Davidov, and A. Rappoport. "ICWSM ? A great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews," in *Proc. AAAI Conf. Weblogs and Social Media*, pp 162-?169, May 2010.

[9] D. Davidov, O. Tsur, and A. Rappoport, "Semi-supervised recognition of sarcastic sentences in Twitter and Amazon," In *Proc.14th Conf. on Computational Natural Language Learning*, pp. 107–116, July 2010.

[10] C. Liebrecht, F. Kunneman, and A. Van Den Bosh, "The perfect solution for detecting sarcasm in tweets #not," in *Proc. WASSA 2013*, pp. 29–37, June 2013.

[11] D. Maynard, and M. Greenwood, "Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis," in *Proc. 9th Int. Conf. Language Resources Evaluation*, pp. 4238–4243, May 2014.

[12] A. Rajadesingan, R. Zafarani, and H. Liu, "Sarcasm detection on Twitter: A behavioral modeling approach," in *Proc. 18th ACM Int. Conf. Web Search Data Mining*, pp. 79–106, Feb. 2015.

[13] A. Joshi, P. Bhattacharyya, and M.J. Carman, "Automatic sarcasm detection: A survey," *arXiv*, Feb. 2016.

[14] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short text classification in Twitter to improve information filtering," in *Proc. 33rd Int. ACM SIGIR Conf. Research and development in information retrieval*, pp. 841–842, July 2010.

[15] C. G. Akcora, M. A. Bayir, M. Demirbas, and H. Ferhatosmanoglu, "Identifying breakpoints in public opinion," in *Proc. First Workshop on Social Media Analytics*, pp. 62–66, July 2010.

[16] M. W. Berry, "Survey of text mining: Clustering, classification, and retrieval", 2004.

[17] B. Pang, L. Lillian, and V. Shivakumar, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. ACL-02 Conf. Empirical Methods in Natural Language Process.*, vol. 10, pp.79–86, July 2002.

[18] M. Boia, B. Faltings, C.-C. Musat and P. Pu, "A :) is worth a thousand words: How people attach sentiment to emoticons and words in tweets," in *Proc. Int. Conf. Social Computing*, pp. 345–350, Sept. 2013.

[19] K. Manuel, K. V. Indukuri and P. R. Krishna, "Analyzing internet slang for sentiment mining," in *Proc. 2nd Vaagdevi Int. Conf. Inform. Technology for Real World Problems*, pp. 9–11 Dec. 2010.

[20] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in *Proc. 52nd Ann. Meeting on Assoc. for Computational Linguistics*, vol. 2, pp. 49–54, June 2014.

[21] F. Jr. Sting, *The meaning of irony.* New York, State University of NY, 1994.

[22] S. G. Shamay-Tsoory, R. Tomer, and J. Aharon-Peretz, "The neuroanatomical basis of understanding sarcasm and its relationship to social cognition," *Neuropsychology*, vol. 19, no. 3, pp. 288–300, May 2005.

[23] C. Burfoot and T. Baldwin., "Automatic satire detection: Are you having a laugh?" in *Proc. ACL-IJCNLP 2009*, pp. 161–164, Aug. 2009.

[24] J. D. Campbell and A. N. Katz, "Are there necessary conditions for inducing a sense of sarcastic irony?," *Discourse Processes*, pp. 459-?480, Aug. 2012.

[25] D. Wilson, "The pragmatics of verbal irony: Echo or pretence?," *Lingua*, Vol. 116, no. 10, pp. 1722–1743, Oct. 2006.

[26] S. L. Ivanko and P. M. Pexman, "Context incongruity and irony processing," *Discourse Processes*, vol. 35, no. 3, pp. 241–279, 2003.

[27] R. Giora, "On irony and negation," *Discourse Processes*, vol. 19, no. 2, pp. 239?264, 1995.

[28] J. Tepperman, D. Traum, and S. S. Narayanan, "Yeah right: Sarcasm recognition for spoken dialogue systems," in *Proc. InterSpeech*, pp. 1838–184, Sept. 2006.

[29] T. Veale and Y. Hao, "Detecting ironic intent in creative comparisons," in *Proc. ECAI*, pp. 765–770, Aug. 2010.

[30] F. Barbieri, H. Saggion, and F. Ronzano, "Modelling sarcasm in Twitter, a novel approach," in *Proc. WASSA*, pp. 50-58, June 2014.

[31] D. Ghosh, W. Guo, and S. Muresan, "Sarcastic or not: Word embeddings to predict the literal or sarcastic meaning of words," in *Proc EMNLP*, pp.1003-1012, Sep. 2015.

[32] Z. Wang, Z. Wu, R. Wang, and Y. Ren, "Twitter sarcasm detection exploiting a context-based model," in *Proc. Web Inform. Syst. Eng. (WISE)*, pp. 77–91, Nov. 2015.

[33] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert, and R. Huang, "Sarcasm as contrast between a positive sentiment and negative situation," in *Proc. Conf. Empirical Methods Natural Language Processing*, pp. 704–714, Oct. 2013.

[34] S. Muresan, R. Gonzalez-Ibanez, D. Ghosh, and N. Wacholder, "Identification of nonliteral language in social media: A case study on sarcasm," *Journal Assoc. Inform. Sci. and Technology*, Jan. 2016.

[35] E. Fersini, F. A. Pozzi, and E. Messina, "Detecting irony and sarcasm in microblogs: The role of expressive signals and ensemble classifiers," in *Proc. IEEE Data Sci. and Advanced Analytics (DSAA)*, pp. 1–8, Oct. 2015.

[36] S. K. Bharti, K. S. Babu, and S. K. Jena, "Parsing-based sarcasm sentiment recognition in Twitter data," in *Proc. IEEE/ACM ASONAM 2015*, pp. 1373-?1380, Aug. 2015.

[37] A. Reyes, P. Rosso, and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," *Data & Knowledge Engineering*, vol. 74, pp. 1–12, Apr. 2012.

[38] A. Reyes, P. Rosso, and T. Veale, "A multidimensional approach for detecting irony in Twitter," *Language Resources and Evaluation*, vol. 47, no. 1, pp. 239–268, Mar. 2013.

[39] A. Joshi, V. Sharma, and P. Bhattacharyya, "Harnessing context incongruity for sarcasm detection," in *Proc. Annu. Meeting Assoc. Computational Linguistics, Int. Joint Conf. Natural Language Processing (ACL-IJCNLP)*, vol. 2, pp. 757?-762, July 2015.

[40] D. Bamman and N. A. Smith, "Contextualized sarcasm detection on Twitter,"in *Proc. AAAI Int. Conf. on Web and Social Media (ICWSM)*, pp. 574–77, May 2015.

[41] L. Derczynski, A. Ritter, S. Clark, and K. Bontcheva, "Twitter part-of-speech tagging for all: Overcoming sparse and noisy data," in *Proc. Int. Conf. RANLP*, pp. 198–206, Sept. 2013

[42] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten "The WEKA data mining software: An update',' *SIGKDD Explor. Newsk.*, vol. 11, no. 1, pp. 10–18, June 2009.

[43] C. Chang and C. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp 20:1–27:27, Apr. 2011.

[44] E. Camp, "Sarcasm, pretense, and the semantics/pragmatics distinction*," *Nos*, vol. 46, No. 4, pp. 587?-634, Dec. 2012.

[45] M. S. Neethu and R. Rajasree, "Sentiment analysis in Twitter using machine learning techniques," in *Proc. 4th Int. Conf. Computing, Commun. and Networking Technologies*, pp. 1–5, July 2013.

[46] L. Breiman, "Random Forest," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Jan. 2001.

[47] S. Attardo, "Irony markers and functions - Towards a goal-oriented theory of irony and its processing" *Rask,*, vol. 12, no. 1, pp. 3–20, 2000.

[48] P. Rockwell. "Vocal features of conversational sarcasm: A comparison of methods," *Journal of psycholinguistic research*, vol. 36, no. 4, pp. 361–369, Sep. 2007.

[49] P. Rockwell, "Empathy and the expression and recognition of sarcasm by close relations or strangers," *Perceptual and Motor Skills*, vol. 97, no. 1, pp. 251–256, Aug. 2003.