ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018

Implementation of the Behavioral Modeling Approach for Sarcasm Detection

[1] Geeta Mehetre, [2] M. B. Kalkumbe
[1][2] MSS College of Engineering, Jalna

Abstract: **Sarcasm is a function of "sarcastic" or "non-sarcastic" labeling. It is a challenging task because there is no pronunciation or sarcasm. Facial expressions in the text However, humans can still see the feeling of severity in the text and the reasons for it. The perception of the friction of the text is an important task for the processing of natural language to avoid the erroneous interpretation of the text in the form of text. The accuracy and durability of the NLP model are often affected by a sense of dishonesty, which is often a mockery. Therefore, it is important to filter the vocal data of training information for various tasks related to NLP. "I'm excited to be called to work all weekend!" It can be classified as a highly positive feeling. However, the fact that negative feeling is implied intelligently through cynicism. The use of cynicism prevails in social networks, sub blogs and forms of electronic commerce. Cramp inspections are necessary for the correct confidence analysis and mining reviews. It can help improve automatic response in the context of client-based sites. Twitter is a small-scale blog platform widely used by people to comment, debate, discuss current events and convey information. Short Message Context the relevant context of the tweets is often identified using the Twitter # (hash-tag) data. It is a rich data repository for implicit sentences that have cynicism.**

Keywords: **- Hashtags, Linguistics, Opinion Mining, Sarcasm Detection, Tweets, Stanford NLP.**

I. INTRODUCTION

The textual data can be divided into two categories, facts and opinions. The facts are objective statements while the opinions are subjective statements. The facts indicate the events that were it happened in the world. Opinions indicate the different feelings, perceptions, observations or points of view about those events. What others think has always been important and interesting information for most of us in the decision-making process. Opinion mining in any company or organization can be considered as-

• When a person wants to buy a phone Look for comments and comments
• A person who has just bought a phone Comments on him writes about his experience
• A phone manufacturer gets customer feedback

Improve your products Adjust Marketing Strategies. When it comes to feelings or emotions, nobody is concerned about the subject of the text, but instead focuses on their positive or negative expressions. People can easily express their opinions on social networking services such as reviews, blogs, social networking sites, as they provide a lot of valuable information. Nowadays, an automatic identification of the feelings is realized that is beneficial for many NLP systems such as review summary systems (SMO), dialogue systems and public systems of media analysis. Mainly, the systems of extraction of existing feeling are based on the identification of the polarity (for example, positive reviews against negative), but there are many types of useful and relatively unexplored feelings like the sarcasm, the irony or the humor. In this document, the feeling of sarcasm has been explored and its detection has been done on Twitter, as a platform. With the recent trend of tag publications using HASHTAGS, some social networking services such as Twitter allow users to add different hash tags to articles / tweets. For this reason, blogs are used as a large set of data for learning and identifying the feeling. In this document, different Twitter tags are used as opinion tags. Different punctuation marks, words and patterns are observed in the text to detect sarcasm.

II. RELATED WORK

Sarcasm and irony are well-studied and emerging concepts in linguistics, psychology and cognitive science [1]. But in the opinion of mining literature, among all these concepts, the automation of sarcasm detection is examined as a difficult problem and has been addressed in a few studies [2]. The tasks of feeling analysis consist of two main steps: (1) Search for different expressions, and (2) determine the polarity (negative, positive or neutral) of the expressed feeling. These steps are usually performed to verify if a sentence conveys a positive or negative meaning. But in this document, sarcastic and non-sarcastic tweets are distinguished to find the polarity of a sentence. It has been proposed that words or phrases of feeling may have different meanings, so that disambiguation of the meaning of the word can improve the analysis of feelings [3]. All the work mentioned identifies expressions of evaluative feeling and its polarity. However, it has been noted that in many cases, simply a sentence cannot be judged as sarcastic or sarcastic without the text or surrounding content. For example, the phrase "Where am I?" it can be assumed to be sarcastic

ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018

only if it is known to be mentioned in a review of a GPS device. In addition, in some cases, analyzing only a few sentences can reveal the presence of sarcasm. Sarcasm in written and oral interaction may work differently [5]. In oral interaction, sarcasm is usually marked by a special intonation [6] or an incongruent facial expression. Since sarcasm is more difficult to understand than a literal statement [5], it is likely that recipients will not interpret the sarcasm and interpret the statements literally. According to Gibbs and Izett [6], sarcasm divides its recipients into two groups; A group of people who understand sarcasm (the so-called group of wolves) and a group of people who do not understand sarcasm (the so-called group of sheep). To ensure that recipients detect the sarcasm in the statement, senders use language markers in their statements. According to Gibbs [6], these markers are clues that a writer can give to "alert the reader to the fact that a sentence is ironic" [6]. On Twitter, the hash tag '#sarcasmo' is a popular marker. The automatic classification of communicative constructions in short texts has become a topic widely studied in recent years. Large amounts of comments, state publications and personal appointments are updated on social networking websites such as Twitter. The process of automatic labeling of the polarity associated with the text as positive or negative can reveal, add or continue in time how the general public thinks about certain things. See Strapparava [14] for an overview of recent research in opinion analysis and opinion mining. An important obstacle to the automatic determination of the polarity of a text (short) are constructions in which the literal meaning of the text is not the intended meaning of the emitter, since many polarity detection systems are based mainly on positive and negative words as markers. The task of identifying these constructs can improve the classification of polarities and provide new perspectives on the relatively new genre of short messages and micro texts on social networks. The previous work describes the classification of irony [9], sarcasm (Tsuret al., 2010), satire (Burfoot and Baldwin, 2009) and humor (Reyes et al.). The work of Reyes et al. (2012b) and Tsur et al (2010). Reyes et al. (2012b) compile an irony corpus based on tweets that consist of the hashtag #irony to form classifiers in different types of characteristics (signatures, unexpected, style and emotional scenarios) and try to distinguish # tweets from iron-tweets that contain hashtags # education, #humor or #politics, achieving F1 scores of approximately 70. Tsur et al. (2010) focus on reviews of products on the World Wide Web, and try to identify sarcastic sentences from them in a semi-supervised manner. Training data can be collected using manual annotations for sarcastic sentences and, in addition, training data can be generated based on phrases recorded as queries. Sarcasm is scored on a scale of 1 to 5. As features, Tsur et al. look at the patterns of these sentences, composed of high frequency words and content words. His system obtained an F1 score of 79 on a product review test, after extracting and recording a sample of 90 phrases classified as sarcastic and 90 phrases classified as non-sarcastic. In the two works described above, a system is tested in an environment

III. PROPOSED ARCHITECTURE

Given a set of tweets, we try to classify each one according to whether it is sarcastic or not. Therefore, from each tweet, we extract a set of characteristics, we refer to a set of learning and we use automatic learning algorithms to perform the classification. The features are extracted in a way that makes use of different components of tweets and covers different types of sarcasm. All the tweets in which we perform our experiments are verified and recorded manually. Since the existing Twitter data set was removed from the server containing 58,609 tweets with the tag "#sarcasmo", we will create another one that will be cleaned up eliminating the noisy and irrelevant ones, as well as those in which Hashtag Fall is used in one of the first two uses of the three described above. With respect to non-sarcastic tweets, we have compiled tweets that deal with different topics and we are sure that they have some emotional content. The data set will contain sarcastic and non-sarcastic tweets. The sarcastic tweets will be compiled by consulting the Twitter API with the hashtag #Sarcasm. To reduce noise, we filter tweets that are not in English, very short tweets (that is, those with less than 3 words) and those that contain URLs. In most cases, URLs refer to photo links. We believe that part of the sarcasm is included in the photo, so we reject them. This data set is used during our experimentation process to optimize the parameters defined for our functionalities.

In the rest of this work, we will refer to this set as "optimization set". The set also contains sarcastic tweets, which are reviewed manually and classified as sarcastic and non-sarcastic. This set will serve as a test and will be used to evaluate the performance of our proposed approach. Therefore, in the rest of this work, it will be called "test set".

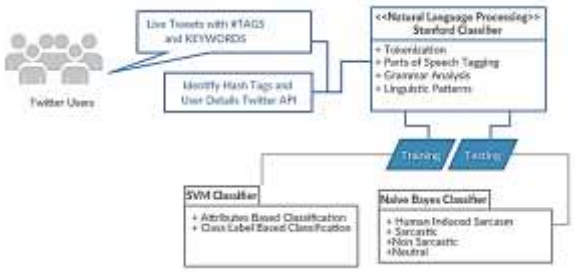


Figure Architectural Diagram of Proposed System

ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018

Framework Sentence Analysis and Grammar Identification

- **Tokenization**

Tokenization [12] is the process of dividing a string sequence into parts such as words, keywords, phrases, symbols and other elements called tokens. Tokens can be single words, sentences or even complete sentences. In the tokenization process, certain characters such as punctuation are rejected. Tokens become the entry for another process, such as text analysis and extraction. Tokenization is mainly based on simple heuristics to separate the tokens following a few steps:

• Tokens or words are separated by spaces, punctuation, or line breaks.

• The blank or punctuation marks may or may not be included if necessary.

• All characters in contiguous strings are part of the token. Tokens can consist of all alpha, alphanumeric or numeric characters.

POS Tagging

Corpus [11] is composed of names or names, which usually appear as the subject or object of a tweet. In the case of dependency grammar, the subject's opinion function has a syntactic relation of the subject verb (SBV) with the predicate of the sentence. The opinion feature of the object has a verbal object dependency (VOB) relation in the predicate. In addition, it also has an object-object dependency relationship (POB) in the prepositional word in the sentence.

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 a greater emotional content than names. Therefore, the positive and negative words that have the associated PoS tag shown in TABLE 1 are re-counted and used to create two other characteristics that we call PW and NW and represent the number of highly positive and highly emotional negative emotional words. We add three additional features by counting the number of positive, negative and sarcastic emoticons. The sarcastic emoticons are emoticons that are sometimes used with sarcastic or ironic expressions (for example, "P"). These emoticons are sometimes used when the person tries to be funny or show that they are just making a joke (that is, when sarcasm is used as a mind). Hashtags also have emotional content. In some cases, they are used to eliminate the ambiguity of the real intention of the Twitter user transmitted in his message. For example, the hashtag used in the following tweet: "Thank you very much for being there for me #ihateyou" says that the user does not want to thank the recipient instead of that he is blaming him for not being there for him. Therefore, we also count the number of positive and negative hashtags.

Punctuation-Related Features

Adjectives, verbs and adverbs have a greater emotional content than names. Therefore, the positive and negative words that have the associated PoS tag shown in TABLE 1 are re-counted and used to create two other characteristics that we call PW and NW and represent the number of highly positive and highly emotional negative emotional words. .

We add three additional features when considering the amount of completely positive, completely negative and completely sarcastic emoticons. The sarcastic emoticons are emoticons that are sometimes used with sarcastic or ironic expressions (for example, "P"). These emoticons are sometimes used when the person tries to be funny or show that they are just making a joke (that is, when sarcasm is used as a mind). Hashtags also have emotional content. In some cases, they are used to eliminate the ambiguity of the real intention of the Twitter user transmitted in his message. For example, the hashtag used in the following tweet that transmits, "Thank you very much for being there when I needed you so much PD: #ihateyou", says that the user does not really want to thank the recipient, instead of blaming him for not being there for him. Therefore, we also count the number of positive and negative hashtags.

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

The excessive use of exclamation marks or question marks, or the repetition of a vowel, especially in an emotional word, may reflect a certain tone that the user tries to show, the tone n 'is not always sarcastic [7]. We believe that these characteristics can be strongly correlated with the number of words in the tweet. Some very short tweets that end with many exclamation marks may surprise more than sarcasm.

Pattern-Related Features

The models selected in the previous subsection and called "common sarcastic expression" [9] are very common, even in spoken language. However, their number is small, they are not unique and most of the tweets in our training and sets of tests do not contain them. However, we deepen and extract another set of characteristics.

We offer more efficient and reliable models. We divide the words into two classes [12]: a first one called "CI" that contains words whose content is important and a second one called "GFI" that contains the words whose grammatical function is more important. If a word belongs to the first

ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018

category, it is lemmatized; otherwise, it is replaced by a certain expression. The expressions [13] used to replace these words are presented in TABLE 2. Classification into classes is done according to the part of the voice tag of the word in the tweet.

POS Tag	Expression
CD	CARDINAL
FW	FOREIGNWORD
UH	INTERJECTION
LS	LISTMARKER
NN, NNS, NNP, NNPS	NOUN
PRP	INTERJECTION
MD	MODAL
PB, RBR, RBS	ADVERBS

SVM Algorithm

Support Vector Machine (SVM) [13, 14] has recently been proposed as an effective statistical learning method for pattern recognition. The SVM based on the theory of statistical learning has many advantages. Unlike previous nonparametric techniques, such as nearer neighbors and the neural network that are based on empirical risk minimization, SVM operates with another principle of induction, called minimization of structural risks, which can overcome the problem of overfitting and the local minimum and obtain better generalization capacity. The Kernel function method is applied in SVM, which does not increase computational complexity; it also overcomes the problem of the curse of dimensionality effectively. SVM has demonstrated a greater capacity of generalization in the space of high dimension and spare samples. Its essence is to map the optimal separation hyperplan that can correctly classify all samples. SVM has proven to be one of the most efficient core methods. The success of SVM [15] is mainly due to its high generalization capacity. Unlike many learning algorithms, SVM leads to good performances without the need to incorporate previous information. Furthermore, the use of the positive defined kernel in the SVM can be interpreted as an embedding of the input space in a high-dimensional feature space where classification is carried out without explicitly using this feature space. Therefore, the problem of choosing architecture for a neural network application is replaced by the problem of choosing a suitable kernel for a Support Vector Machine.

The support vector machine has shown power in the binary classification. It has a good theoretical base and a well-mastered learning algorithm. It shows good results in the classification of static data. The only disadvantage is; It consumes time and memory when the size of the data is huge. SVM can be used to solve linear and non-separable separation problems.

Algorithm

Support Vector Machine uses Kernel Functions to map training data in the function space. Consider the mapping of X and Y, where "x X" is an object and "y ∈ Y" is a label. Therefore, the classifier is given as $y = f(x, \alpha)$, where α gives us the parameters of the functions. In most circumstances, the data set can be linearly separable. For this, we need a simple classifier,

Here w and b are taken from the training set 'x'. The decision function is given as

The data includes good and recognized messages of sarcasm. The information that contains the details of the profile and the details of the message is stored in the ".lsv" file. This information is used to train the Support Vector Machine. In the query selection phase, the data set file with profile details and message details are read through the name of the route. In the query selection phase, the text file containing the profile details and the message details are read through the path name.

Naive Bayes Classifier

Naive Bayes is a probabilistic classifier, which means that for a document d, all classes $c \in C$ the classifier returns the class c that has the maximum posterior probability given the document.

The Naive Bayes classifier is a simple probabilistic classifier that is based on Bayes' theorem with strong and naive assumptions of independence. This is one of the most basic text classification techniques with several applications in the detection of electronic mail, classification of personal messages, categorization of documents, detection of sexually explicit content, detection of languages and detection of feelings. Despite the naive design and simplified assumptions that this technique uses, Naive Bayes works well on many complex problems in the real world.

Although it is often surpassed by other techniques such as power trees, random forests, Max Entropy, Support Vector Machines, etc., the Naive Bayes classifier is very effective since it is less expensive in computing (both CPU and memory) and requires a small amount of training data. In addition, the training time with Naive Bayes is much shorter compared to alternative methods?

The Naive Bayes classifier is superior in terms of CPU and memory consumption as shown by Huang, J. (2003), and in many cases their performances is very similar to the more complicated and slow techniques.

A naïve bayes classifier [15] is a simple probabilistic model based on the Bayes rule with a strong hypothesis of independence. The Naïve Bayes model implies a simplified hypothesis of conditional independence. This is given a class (positive or negative), the words are conditionally independent of each other. This assumption does not significantly affect the accuracy of the text classification, but makes the classification

ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018

algorithms very fast and applicable to the problem. In our case, the probability of maximum probability that a word belongs to a certain class is given by the expression:

$$P(x|c) = \frac{count of x in tweets of class c}{total number of words in class c}$$

Here, the xi are the individual words of the post tweet. The classifier delivers the class with the maximum likelihood a posteriori. We also eliminate duplicate words from tweets, do not add any additional information; This type of Naive Bayes algorithm is called Bernoulli Naïve Bayes. It has been found that the inclusion of the presence of a word instead of the count marginally improves performance when there are a large number of examples of training.

Experimental Setup

To form an algorithm for detecting sarcasm, first we need to train that algorithm and we require data for that. Classification is a directed learning job, which means, for the
classifier to know the difference between different sentences, some sentences labeled as sarcastic and others labeled as no sarcastic are needed. It can be done by using an online corpus which contains various sarcastic sentences, for example reviews, comments, posts etc. and labeling is done. But this is very monotonous exercise in case of large data set. Another option is to make use of the Twitter API to club tweets with the label #sarcasm or #sarcastic, these will be the sarcastic tweets, and others that don't have such label, will become non-sarcastic tweets.

Our machine is trained using positive and negative dataset from https://www.cs.uic.edu /~liub /FBS /sentiment-analysis.html#lexicon

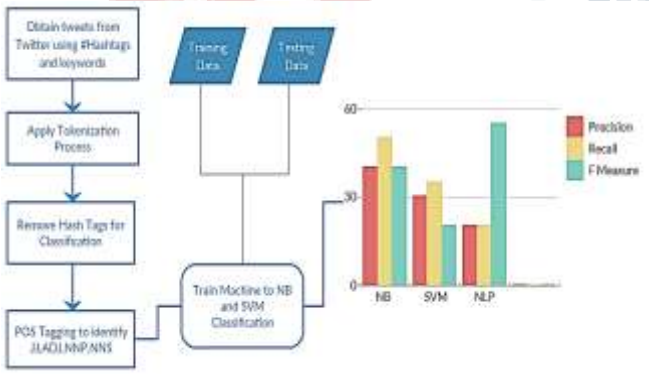


Figure Functional Flow Diagram

In Twitter, a message can be of about 140 characters. Except the normal text, a twitter message can contain references to other users (@<user>), hashtags (#hashtag) and URLs. For example, : "@personA Check out @personB for amazing ideas :) http://xxxxxx.com #happy #hour"[4]. So for building the corpus of sarcastic (S), negative (N) and positive(P) tweets, the annotations that tweeters assign to their tweets using hash tags are used. Twitter API is used to collect tweets that include hashtags of sarcasm (#sarcastic, #sarcasm), direct positive sentiment (e.g., #happy, #joy, #lucky, #amazing, #exiting), and direct negative sentiment (e.g., #sadness, #angry, #frustrated, #bad, #fail) [5]. Also, automatic filtering is applied to remove quotes, spam, duplicates, and tweets written in languages other than English. The advantage of using Twitter API is that we can have enough samples to fulfill our requirement. Every day people write tweets, use sarcasm that can be easily collected, clubbed and stored in a database. But there's a drawback in collecting data from Twitter, that is, the data is little noisy! People also use the #sarcasm hash tag to show that the tweet is sarcastic, but a Human cannot simply guess or assume that the tweet is sarcastic without the label #sarcasm. So for this we need to pre-process the data i.e. cleaning up the data. For doing this, all the tweets which contain Non-ASCII characters, link to other tweets and non-sarcastic behavior, are removed. After that all the hash tags and all occurrences of the word sarcasm or sarcastic are removed from the remaining tweets. And still if the tweet is at least 3 words long, it is added to the dataset [6]. The above is done to remove all the noise.

In information retrieval with binary classification, precision (also called positive predictive value) is the fraction of retrieved instances that are relevant, while recall (also called sensitivity) is the fraction of the relevant instances that are retrieved. Precision and recall are therefore based on understanding and measuring relevance. In simple terms, high accuracy means that an algorithm returns significantly more relevant than irrelevant results, while a high recall means that an algorithm has yielded the most relevant results.

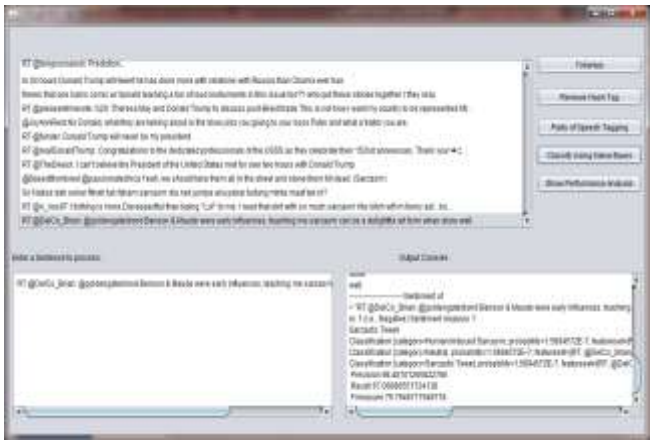
The most important category measurements for binary categories are:

Precision	Recall	F Measure
$P = TP/(TP + FP)$	$R = TP/(TP + FN)$	2*P*R/(P+R)

Screen 1) Tweets Extraction

ISSN (Online) 2456 -1304

International Journal of Science, Engineering and Management (IJSEM)
Vol 3, Issue 2, February 2018



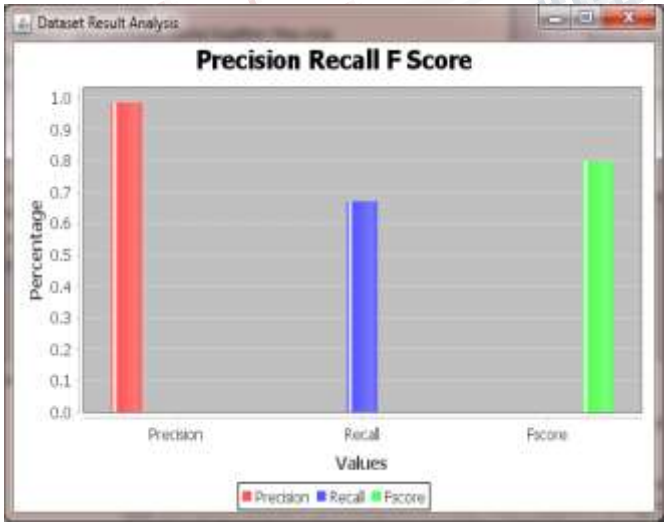
Screen 2) Tweets Classification Process

Table1.1 Confusion Matrix

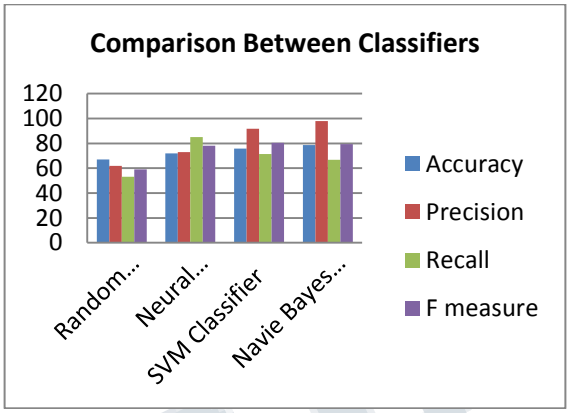
Confusion Matrix	Predicted True	Predicted False
Actual True	16	8
Actual False	9	11

Table1.2 Summary and Result Comparison with Existing System

Classifiers	Accuracy	Precision	Recall	F measure
Random Forest Classifier	67.00%	62.00 %	53.00 %	59.00%
Neural Network	72.00%	73.00%	85.00 %	78.00%
SVM Classifier	75.80%	91.90%	71.39 %	80.36%
Naive Bayes Classifier	78.66%	97.83%	66.91 %	79.47%



Graph 1.0 Generated by Application



Graph 2.0 Comparison Graph

IV. CONCLUSION & FUTURE SCOPE

The detection of sarcasm is a really fascinating topic. Evaluate different types of characteristics to extract feelings, including feelings, words, patterns and n-grams, confirming that each type of characteristic contributes to the classification of frame feelings. In this work, we propose a new hybrid method to detect sarcasm on Twitter. The proposed method uses the different components of the tweet. Our approach uses Partof-Speech-tags to extract models that characterize the level of sarcasm of the tweets. In the future, these methods can be applied to the automated grouping of feelings and rules of feeling dependence and can be developed to detect other forms of non-literary feelings, such as humor. Another thing that could be interesting is to see the sentences separately and classify how sarcastic they are, for example, this sentence is 45% sarcastic. However, finding a general solution to this problem is very difficult. But building models that can find irony in a specific type of text is something that could be easier. E.g. the classifier that only classifies Twitter tweets and one that only classifies news articles is the example. This is because the text differs a lot depending on the context and the language used. Also from the results it is shown that the classifiers work differently in the data sets. When observing our results, it is clear that vocabulary is the most discriminatory characteristic to find irony in the data sets that have been used. Therefore, this is important for future work to test other functions than what we did together with the vocabulary to see if you can give the model more precision. Finally, we recommend that more research be done in the area in this area. There is still a long way to go until an adequate irony detector can be used in situations in general.

REFERENCES

1. Carvalho, P., Sarmento, S., Silva, M. J., and de Oliveira, E. 2009. Clues for detecting irony in user-generated contents: oh...!! it's "so easy" ;-). In Proceeding of the 1st international CIKM workshop on Topicsentiment analysis for mass opinion (TSA '09). ACM, New York, NY, USA, 53-56.

2. Clark, H. and Gerrig, R. 1984. On the pretence theory of irony. Journal of Experimental Psychology: General, 113:121–126. D.C.

3. Davidov, D., Tsur, O., and Rappoport, A. 2010. SemiSupervised Recognition of Sarcastic Sentences in Twitter and Amazon, Dmitry Proceeding of Computational Natural Language Learning (ACL-CoNLL).

4. Derks, D., Bos, A. E. R., and Grumbkow, J. V. 2008. Emoticons and Online Message Interpretation. Soc. Sci. Comput. Rev., 26(3), 379-388.

5. Gibbs, R. 1986. On the psycholinguistics of sarcasm. Journal of Experimental Psychology: General, 105:3–15.

6. Gibbs, R. W. and Colston H. L. eds. 2007. Irony in Language and Thought. Routledge (Taylor and Francis), New York.

7. Kreuz, R. J. and Glucksberg, S. 1989. How to be sarcastic: The echoic reminder theory of verbal irony. Journal of Experimental Psychology: General, 118:374-386.

8. Kreuz, R. J. and Caucci, G. M. 2007. Lexical influences on the perception of sarcasm. In Proceedings of the Workshop on Computational Approaches to Figurative Language (pp. 1-4). Rochester, New York: Association for Computational. LIWC Inc. 2007.

9. The LIWC application. Retrieved May 10, 2010, from http://www.liwc.net/liwcdescription.php.

10. Nigam, K. and Hurst, M. 2006. Towards a Robust Metric of Polarity. In Computing Attitude and Affect in Text: Theory and Applications (pp. 265-279). Retrieved February 22, 2010, from http://dx.doi.org/10.1007/1-4020-4102-0_20.

11. Pak, A. and Paroubek, P. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining, in 'Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)' , European Language Resources Association (ELRA), Valletta, Malta

12. Pang, B. and Lee, L. 2008. Opinion Mining and Sentiment Analysis. Now Publishers Inc, July.

13. Pennebaker, J.W., Francis, M.E., & Booth, R.J. (2001). Linguistic Inquiry and Word Count (LIWC): LIWC2001 (this includes the manual only). Mahwah, NJ: Erlbaum Publishers

14. Strapparava, C. and Valitutti, A. 2004. Wordnet-affect: an affective extension of wordnet. In Proceedings of the 4th International Conference on Language Resources and Evaluation, Lisbon.

15. Tepperman, J., Traum, D., and Narayanan, S. 2006. Yeah right: Sarcasm recognition for spoken dialogue systems. In InterSpeech ICSLP, Pittsburgh, PA.

16. Utsumi, A. 2000. Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony. Journal of Pragmatics, 32(12):1777– 1806.