Automatic Sarcasm Detection using feature selection

[1]Paras Dharwal,[2]Tanupriya Choudhury,[3]Rajat Mittal,[4]Praveen Kumar

[1,2,3,4]Amity School of Engineering and Technology, Amity University Sector 125, Noida, Uttar Pradesh

[1]letmynamebexyz@gmail.com, [2]tchoudhury@amity.edu, [3]rajatmittal100@gmail.com,[4]pkumar3@amity.edu

***Abstract*-Sarcasm is an expression of humor, mockery or criticism with the help of ironic remarks that seems positive. Sarcasm detection in sentiment analysis is essential for understanding the emotions and thoughts of the people. Automatic sarcasm detection refers to the detection of sarcasm in the text written in natural language. Various natural language processing techniques carry out this purpose. Recognition of Sarcasm is of extraordinary significance and is valuable to numerous NLP applications such as Opinion Mining andmultiple advertisings. Sarcasm detection is a complex task, because of the challenges involved in determining the nature of the sarcasm bearing text. Automatic sarcasm detection is thus, one of the hardest challenges in Sentiment Analysis, including complex linguistic analyzing and machine learning methods. This paper focuses on various sarcasm analyzing techniques employed for filtering of sarcastic statements from a text and the use of Automatic sarcasm detection in the categorization of tweets and product review texts.**

Keywords: Sarcasm, NLP (Natural Language Processing), opinion mining, linguistic analyzing, tweets, automatic

I. INTRODUCTION

Merriam Webster[1] defines Sarcasm as a mode of satirical wit depending for its effect on bitter, caustic, and often ironic language that is usually directed against an individual.Since sarcasm is for the emotions and thoughts of the people, it becomes a significant part of sentiment analysis and hence, becoming essential to understand the true meaning of the text. Usually, sarcasm while talking could be easier to identify because detection becomes simpler through the tone,pitch, expressions of the speaker whilein the text these gestures and expressions are missing which makes the detection of the same becomes harder.A sentence may seem to be positive, however implicitly it could be negativeor ironic. E.g., if a person is writing a review comment on a coffin selling as “very comfortable for sleeping”then this type of text must be regarded as sarcastic whereas the same comment about a sleeping mattress cannotbe putinto thecategory of sarcasticcomments. It becomes essential and challenging to identify the true nature of the sentence. Automatic sarcasm detection deals with the detection of sarcasm in texts (in natural language) via computational methods. [1]The primary goal of this research paper is to understand different methodsfor Automatic sarcasm detection. Further, this paper describestechniques usedforcollectively studying the various approaches available in Automatic Sarcasm Detection Field.

II. DATA PROCESSING AND RELATED WORK

In General Sciences and Linguistics, the study of sarcasm and irony happens through the speaker’s way of expressing himself. Sarcasm Detectionisutilized mainly as a part of interpersonal organizations and micro-blogging sites, where individuals deride or mock in a way that makes it troublesome, notwithstanding for people to tell if what is said is what is implied.On Twitter, the distributive filtering of the textual statements is done in two ways: i.e.,by the expressions that the text implies, the second way is the distribution of the tweets into three majortypes,i.e., the positive, negative and the sarcastic statements or tweets,i.e.,by the polarity of the tweet. The Amazon[3] review processing is slightly different.The detection of sarcasm is dependent on the text associated with the review. TheOmission of special cases such as, “It's” could refer to “its” as well as "it has" and "it is". Therefore,neglecting such cases is essential. Further, development of sentiment features to find the polarity (negative or positive) of the review, takes place to analyze the corpus with sarcastic snippets. Previous works describe the classification of irony (Reyes et al., 2012b), sarcasm (Tsur et al., 2010), and humor (Reyes et al., 2012a). The idea here is to find a corpus that consists of tweets having #irony and differentiating form any other hash tags getting a F-Score of around 70.

III. DATA ANALYSIS AND ARCHITECTURE

To make the designed algorithm understand the basic functionality that it provides,weneed to train it. Similarly, in the case of automatic sarcasm detection, we will need to train the algorithm to know the difference between the sarcastic and the non-sarcastic comments and tweets. For implementation purposes, we use a corpus which already has a list and viewing of the sarcasticsentences present in the corpus. For example, one can have an online corpus, consisting of a large variety of the phrases and words that are frequently used in the sarcastic comments. Labeling of the posts, review comments, tweets is also a possibility. Emoji and the emoticons associated with the reviews and comments also can tell a lot about the text. [3] A comment or review with a winking emoji or Lenny face or laughing emoji is highly likely to be sarcastic or ironic and thus, further narrowing down the lists of posts that require a check for sarcasm detection.E.g.in Twitter we have a Corpus that is analyzedand categorized into sarcastic, negative and positive tweets we use the following annotations for each of the emotions positive (P,)negative (N) and sarcastic(S).Twitter API is useful when it comes to analyzing and identifying the hash

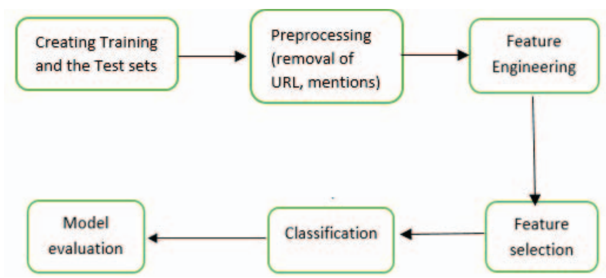
978-1-5386-1144-9$31.00 ©2017 IEEE

tags. Following sentences are categorized as sarcastic or not on the basis of prediction by thesarcasmdetector as shown in table 1.The sarcasm score ranges from -100 to 100.

Sentence	Prediction	Sarcasm Score
I just love the morning alarms!	Sarcastic	+56
This sentence is just for testing whether long texts are considered sarcastic or not.	Non-sarcastic	-25
Nice perfume, must you marinate in it!	Sarcastic	+21
I want to be a doctor	Non-sarcastic	-29
How are you feeling today?	Non sarcastic	-14

Table 1: prediction of sarcastic text

The architecture for the Automatic sarcasm Detection is shown



in the Fig 1.

Fig 1: proposed architecture

IV. FEATURE ENGINEERING:

Feature Engineering is involved in studying and analyzing the text under various categories. The different steps used for this are:

IV.A. Tokenization: Tokenization is one of the main features of lexically analyzing the text. Here the aggregation of the sequence of characters takes place. Decluttering the words of the sentence into tokens is very helpful in individual identification [2]. These counts of tokens are very helpful. A token could be a complete group of words or a whole paragraph, but most frequently, we use words as tokens. E.g.using Hashtag-Tokenizer to separate the hashtags that contain connected words. e.g., splitting #Sarcasticirony into #sarcastic and #irony.

IV.B. Stemming: It is a common method in Natural Language Processing (NLP). The main motive of this is to reduce the repetition of words by dropping the suffix of the words to arrive at a basic form of the word. Hence, transforming the text into a more accurate form that is easily analyzable. The basic idea behind stemming is that we group the words with common or close meanings together and thus try to enhance the efficiency of NLP. To increase the efficiency, applications should be made of stemming to group these words into a single term. For example, the words: depressing, depressed, depress, depresses, depression, etc. are having a common stem here which is "depress" accompanied by the suffixes as "ed," "ing" etc. It is not necessary that a suffix would have a meaning or not.

IV.C. n-grams: n-grams is the group of co-occurring n-items in a sequence, which can be speech or text, used largely for NLP purpose. Example: "You should definitely go there" is a sentence in the form oftext. Unigram, bigram and trigram are shown in Fig 2. Here the n-grams we have are of four forms, "you should" to "go there."

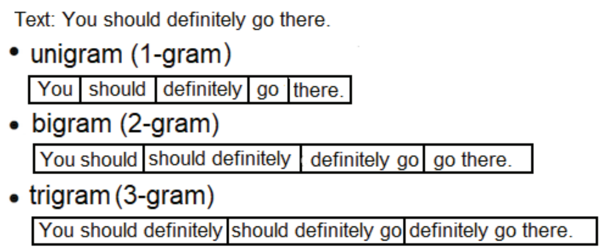


Fig 2: n-grams for the sentence

One can easily achieve this by moving one word at a time to accomplish the next bigram. Similarly, we can produce unigram, trigram or four grams or five grams and so on for N=1, N=3, N=4, N=5 and so on.

To extract grams for each the text goes through tokenization, and through stemming, is uncapitalized and then finally every n-gram is added to the dataset as shown in Fig 3.

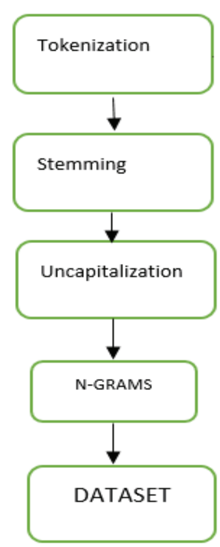


Fig 3. flowchart for feature engineering

IV.D. Sentiments:

There is an emotion associated with each post or tweet we see. Some can be positive while the others are negative. The sarcastic posts we see are more negative as compared to the non-sarcastic posts. This makes it very important for analyzing each part of the text. Sentiment analyzers that are available analyze each token very carefully and thus make the categorization of text easy[4].

Sarcastic comments consist of a positive part, which is ironic. For example, “the coffin is very comfortable to sleep in #sarcasm”. Thus, it becomes essential to analyze each of the parts of the sentence very carefully. We have an analyzer called SentiWordNet taking up each word we have in our sentence, and for every single word, it has a rated value of being positive or negative. Also, programming languages provide various tools for analyzing the texts and thus helps in opinion mining. E.g., NTLK (natural language toolkit) is a python library, which is helpful while working with natural language processing. This consists of more than fifty variant lexical assessing tools that can be greatly helpful to analyze the data. Text blob which is another library, works with the data that is in the form of the text. It provides built-in automatic ranking functions, which are very helpful while categorizing the sentiments in texts.

IV.E. Recognition of Pattern:

Pattern recognition is very important when it comes down to recognition at semi-supervision level. This is used for analyzing sarcastic tweets and for the Amazon review comments. Achieving it is possible with the help of creating pattern among the words and categorizing words into High-Frequency Words (HFW) and content words (CWs). High-Frequency Words appear mostly and have high chances of occurrence in the text. These consist of the single punctuation characters because their frequency of occurrence is relatively high as compared to the other characters [5]. Each Pattern has different slots for different types of words. E.g., for CWs the slots available range from 1-6 whereas for HFWs the same is 2-6. This allows one sentence to form multiple types of repetitive designs. While performing the feature selection removal of the highest appearing pattern (occurring is most tweets) and lowest appearing patterns (occurring in the least tweets) takes place. Ratings are given to each of the patterns according to the category which they are foundin.

1. Exact Match: all the words in the patter nappear contiguously in the right order without any additional components.

α: Sparse Match: all the words in the patternare found, but there are some additional words present.

γ.n/N: Incomplete match: only n>1 of the total N patterns appear in the sentence while some non-matching words can be insertedin between. A special condition is that one word in this must be the high-frequency word.

0: No match: Nothing or only a single pattern component appears in the sentence.

V. FEATURE SELECTION:

In feature selection, a subset is selected for use in classification. On the basis of features,categorization of the classes is done so that they can be differentiated from each other. Each class has different features than the other.Redundant features are often present in the dataset. Redundant features are those which do not contribute in distinguishing the classes from each other. These can thus be removed without incurring much loss of information.

Term frequency-inverse document frequency (TFIDF): it is a part of information retrieval and shows that how important a word is to a corpus. The TFIDF value for a word in a document increases with the increase in the number of times the word appears in the document. At the same time, the words which are available in all the documents tell a very little or no information about the document.

tf-idf = tf(t, f) ×−idf(d,D)

Term frequency tf(t,f) is chosen simply by using the frequency f(t,f) and t, here is the number of time the word occurs in document d. Idf is a measure of the significance of the word. It is quite obvious that occurrences of words like “to”, “for” etc. is quite high in documents [8]. Hence these words are given less importance.The following equation diminishes the importance of such words in different documents by dividing the logarithm of the number total number of documents N by the number of documents d in D which contains the word t.

Chi-square: is a statistical method used to test the independence of two categorical variables. If two variables in a document are similar, then a low result is obtained, and the higher result would be obtained for words that differ.Defination: If ν

$$idf(t,D) = \log \frac{N}{|\{d \in D : t \in d\}|}$$

independent variables x are each normally distributed with mean μ and variance σ_i^2, then chi-squared is defined as:

Here if the chi-squared method will help in finding out the

$$X^2 = \sum_{i=1}^{\nu} \frac{(x_i - \mu_i)^2}{\sigma_i^2}$$

words which have identical distribution in documents and thus irrelevant.

VI. CLASSIFICATION

Algorithms are highly useful and needed for automatic detection of the sentiment related to the text. Positive, negative and sarcastic nature of a text is analyzable by using the features of algorithms to analyze our text. Here we have different types of classifications: Logistic regressions (LogR) and SVM[14].

VI.F.A. Support Vector Machines (SVMs): These are the new machine learning technique. One feeds labeled training data set to the SVM to get an optimal hyperplane[4], which can help in classification and categorization of new data. On both sides of the hyper plane, we have instances (which consists of data points in space belonging to the two classes). After finding a hyper plane, the distance maximizes (distance between the nearest data point of each class and the hyper plane) between the two classes and the hyper plane[6].

Fig 4 shows the 2D plane and two classes with data points in space. One discovers a separating straight line between the two classes. We have multiple lines as a solution for this.

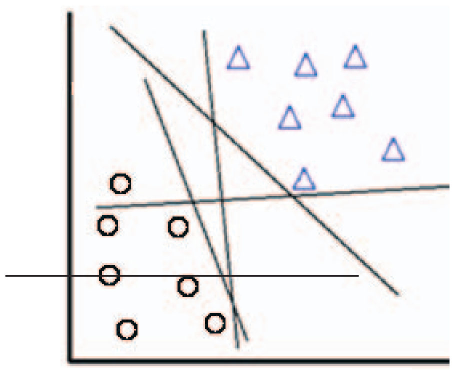


Fig 4: Possible Hyper planes

However, the optima; hyper plane is the one which maximizes the margin as shown in fig 5.

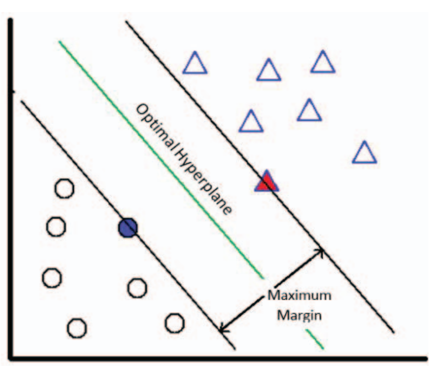


Fig 5: Optimal hyper plane

VI.F.B Logistic regression: Logistic regression is a statistical method used prominently in machine learning. [7]

It is a problem involving binary classification ,i.e., which can take two values. Measurement of the outcome is done with the help of dichotomous variables. E.g., Email: spam or not. In both these cases, two values for each case are possible. The First case can have two outcomes,i.e., spam and not spam (dichotomous).
Bamman and Smith used logistic regression for binary (sarcastic or non-sarcastic) classification of tweets. For example, while validating models bases on feature-type categorization takes place by capitalization feature of Twitter.
Logistic regression for binary classification is dependent on

Fitting the following equation:

$$\sigma(t) = \frac{e^t}{e^t + 1} = \frac{1}{1 + e^{-t}}$$

Where $t=\beta_0+\beta_1 x_1+\beta_2 x_2+\ldots+\beta_m x_m$.

Regression algorithms deal with plotting the data coordinates alongside their yield value in space and then finding a function that can estimate the output value of a coordinate[13].

VI.F.C Naive Bayes: Naive Bayes Classification is very famous classifier because of the simplicity of the use. It is the most basic form of text categorization. Naive Bayes is a conditional probability model. The classifier makes the use of Bayes theorem and makes naive estimations. It exists for spam filtration purposes also. One downside of this is that it is not very trustworthy as probability outputs can be inaccurate sometimes.

x is a vector value representing n variables

(x= x1+x2+x3+……+xn).

P() is the probability function, c is the classes.

VII. FINAL EVALUATION AND ANALYZING:

The main purpose of the sarcasm detection framework is to see that is the system is well defined and can easily detect and categorize the text into positive, negative and sarcastic. Thus

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

testing of the model is essential. For this purpose, we divide the entire dataset into three parts, i.e., Training set, Validation set and Testing set. The training set consists of the data with the help of which the model is trained. This dataset consists of input as well as the correct output. The dataset is referred to as Gold Standard. The validation set is used to find the model which best fits the training set by using the algorithms that give the best results. It shows the extent to which the model has been trained. The third is the testing set. This consists of the real world data, and the model is tested against this data and results are found. [10]This is done by checking the accuracy of the proposed model.

This is to testify, whether the framework can identify ironic or sarcastic nature of the text with the help of tags and in those where tags are not present. Cross-examining helps in doing this. The primary purpose of this is that we have a dataset that is present for sarcasm detection. After the analysis, we see that if the results produce were accurate or not. A certain accuracy level is required for passing the framework for making advantageoususe of it. Otherwise, if the accuracy rate is less, it means that still further improvements and changes are required. F-Score is the term, used to describe the level to which the code is accurate.F-scores for various features are as shown in the table 2:

Feature	F-Score
n-grams	0.56
sentiments	0.41
topics	0.35

Table:2 F scores

This is highly useful when we have more texts from a particular category. E.g., we have more positivetexts than negative, and then this is useful. Precession is very important while evaluating the piece of text. Precision here defines the correct number of sarcastic texts detected between the total and the actual no of texts that are present. The ratio of these determines the value of precision in detection. Finally, observations from these sarcastic comments, reviews, and posts are more expressive regarding emotion as compared to the non-sarcastic posts, whether the nature of sarcasm associated with it is positive or negative.

VIII. CONCLUSION

Over the years, researchers have been carrying out significant work in the field of sarcasm detection. The growth in the area of automatic sarcasm detection is prominent. This paper discusses the methods that are used for Automatic Sarcasm detection. Although these methods can help in sarcasm detection, they are not entirelyefficient in detection of sarcasm and categorization of the texts. As discussed in Section IV, being solely dependent on feature engineering is not a good approach. [12]The use of n-grams is not sufficient for accurately classifying. Although combining these with other methods can increase the accuracy to a great extent.

Naïve Bayes method also does not lead to higher precision, and the chances of poor results are quite high in this. Methods like, SVM need to be more precise in their analysis because these can provide more accuracy. However, the main issue with SVM is both the speed and the size (both in training and testing). Logistic regression also can provide excellent results but not as efficient as SVM[16]. Logistic regression as discussed relies on the independence of the data points. Also, Logistic regressions cannot be used for continuous outcomes. Development of new techniques and methods can provide users with more than one feature and better results. Many other implementations still need to be embedded into the detection frameworks like understanding how the same words while being used in different sentences can be considered to be sarcastic and non-sarcastic.[11] These type of implementations can further help in minimizing the effort in the filtering of sarcastic texts from other. Although Sarcasm detection is a hot topic in sentiment analysis, there is still a lot to be done, especially from the social media In the future; further advancements would be included in the Automatic detection Systems that can help to understand the behavior, emotion, and opinion of the people in a more explicit way.

REFERENCES:

[1] Rilof, E., A.Surve, P, De Silva, L., Gilbert, N., & Huang, R. (2013). Sarcasm as contrast between positive sentiment and negative situation.

[2] B. Liu Sentiment analysis and opinion mining SLHLT 5 (1) (2012), pp. 1-167.

[3] Amazon General review creation guidelines. http://www.amazon.com/gp/community-help/customer-reviews-guidelines.

[4] Hastie et al., 2010Hastie, T., Tibshirani, R., and Friedman, J. (2010). The Element of Statistical Learning. Springer.

[5]Bing Liu, Lei Zhang: A survey of opinion mining and sentiment analysis. In Mining Text Data Springer US. 2013.

[6]Francesco Barbieri, HoracioSaggion, and Francesco Ronzano. 2014. Modelling Sarcasm in TwiŠer, a Novel Approach. ACL 2014 (2014), 50.

[7] Gibbs Jr, R.W., Colston, H.L.: Irony in language and thought: A cognitive science reader. Psychology Press (2007)

[8] Sentiment analysis in social networks by FA pozzi, elisabettafersini, enza sentiment analysis in social networks

[9]http://www.thesarcasmdetector.com/about.

[10]https://en.wikipedia.org/wiki/Support_vector_machine.

[11] https://en.wikipedia.org/wiki/Logistic_regression.

[12]https://www.aaai.org/index.php/ICWSM/ICWSM15/paper.

[13] https://www.merriamwebster.com/dictionary/sarcasm

[14] http://nlp.stanford.edu/IRbook/html/htmledition/tokenization- 1.html

[15]https://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html

[16]https://ufal.mff.cuni.cz/~zabokrtsky/courses/npfl104/html/feature_engineering

[5]. Bing Liu, Lei Zhang: A survey of opinion mining and sentiment analysis. In Mining Text Data. Springer US. 2013. p. 415-463.

[6]. Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014. Modelling Sarcasm in Twi er, a Novel Approach. ACL 2014 (2014), 50.

[7]. Gibbs Jr, R.W., Colston, H.L.: Irony in language and thought: A cognitive science reader. Psychology Press (2007)

[7]. Identifying Smaranda Muresan Nina Wacholder School of Communication & Information Rutgers .

[8] http://www.thesarcasmdetector.com/about.

[9] http://nlp.stanford.edu/IRbook/html/

htmledition/tokenization- 1.html

[10]https://en.wikipedia.org/wiki/Support_vector_machine.

[11].https://en.wikipedia.org/wiki/Logistic_regression.