International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

SARCASM DETECTION IN ONLINE SOCIAL NETWORKS USING DEEP LEARNING ALGORITHMS

[1]Dr.R.Josphineleela. M.E, Phd [2]Hema Swetha S [3]Ponnaa [4]Mu,Hemasrivalli H

[1] *HoD, Department of Information Technology, Panimalar Institute of Technology ,*

[2,3,4] *Student, Department of Information Technology, Panimalar Institute of Technology ,*

Abstract - Sarcasm is a sophisticated form of irony widely used in social networks and micro-blogging websites. It is sometimes wont to convey implicit data inside the message an individual transmits. Sarcasm can be used for various functions like criticism or mockery. However, it's onerous even for humans to acknowledge. Therefore, recognizing sardonic statements is terribly helpful to enhance automatic sentiment analysis of information collected from micro blogging 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 tend to propose a pattern-based approach to observe humor on Twitter. We propose four sets of options that cowl the various forms of humor we tend to defined. We use those to classify tweets as sardonic and non-sarcastic. Our projected approach reaches associate accuracy of eighty-three .1% with a precision equal to 91.1%. We conjointly study the importance of every of the projected sets of options and measure its side price to the classification. In particular we tend to emphasize the importance of pattern-based options for the detection of sardonic statements.

I. INTRODUCTION

Sarcasm is part of human nature and perhaps an evolutionarily noble entity. It is the routine of remarks that undoubtedly refer to the opposite of what the individuals say and make in order to miffy someone's feelings or to disparage something in a hysterical way. Sarcasm has a two-faced quality: it's both comical and means. Machine learning methods and algorithms gave a new way for sentiment analysis particularly sarcasm detection by providing a set of algorithms and procedures.

Sarcasm is "a sharp, bitter, or cutting expression or remark; a bitter gibe or taunt". Sarcasm is sometimes used as merely a synonym of irony, but the word has a more specific sense: irony that's meant to mock or convey contempt. This meaning is found in its etymology.

A field that takes ideas from machine learning and applies them to text data. Your email spam filter is an application of NLP; there is a learning algorithm that learns how to differentiate a spam email from a regular email by looking at the text content of the email. It had just come out in the news that the U.S. secret service was looking for a sarcasm detector to improve their intelligence coming from Twitter and I was curious. If you look at a spam filter algorithm, the features that will be most relevant to the classification of emails will be certain keywords: Notspam, Free access or Enlarge your ... for instanceI thought sarcasm is hidden in the tone and the ambivalence of the sentence. Merriam-Webster defines sarcasm as the use of words that mean the opposite of what you really want to say especially in order to insult someone, to show irritation, or to be funny. So to detect sarcasm properly a computer would have to figure out that you meant the opposite of what you just said. It is sometimes hard for Humans to detect sarcasm, and Humans have a much better grasp at the English language than computers do, so this was not going to be an easy task.

II. EXISTING SYSTEM

. The extraction method of sarcastic sentences in product reviews For our method, we collected sarcastic sentences to analyze them in advance. We manually labeled 70 sentences as sarcastic sentences from 10,000 reviews. We generated extraction rules on the basis of the analysis of the sentences. The rate of sarcastic sentences contained in reviews was low (70/10,000).

However, 21 sarcastic sentences appeared in 233 reviews with 1-point, which is the worst point in this review dataset. This result shows a significance of sarcasm extraction even if the number of sarcastic sentences in reviews is small. In the experiment, we compared our method with a baseline based on a simple rule. As a result,

Our method outperformed the baseline However, some approaches to extract sarcastic sentences have, such as Riloff's method. Comparison with state-of-the-art methods is important future work to evaluate our method. In Addition, the accuracy of our method was insufficient, especially the precision rate. The result is due to the lack of analysis. Although we analyzed sarcastic sentences in our data, the data contains only 70 sarcastic sentences. Collecting new sarcastic sentences and analyzing the sentences manually are important.

III. PROPOSEDSYSTEM

Manual analysis of numerous sentences is costly. Therefore, generating rules automatically becomes necessary .In this we are going to use Large dataset and many classifiers to get the maximum accuracy. The classifiers like SVM ,Random Forest , Logistic Regression , Decision Tree, Neural Networks, Naive Bayes In this project we use 21 special features along with usual unigrams and bigrams for classification. So that the accuracy of our sarcasm detection will be improved. The fundamental commitments of this project are:

(1) We propose a committed GAN-based methodology that successfully takes in the planning from genuine videos to animated videos utilizing unpaired picture sets for preparing. Our strategy can create top notch adapted kid's shows, which are considerably in a way that is better than cutting edge strategies. At the point when animation pictures from singular specialists are utilized for preparing, our strategy can repeat their styles.

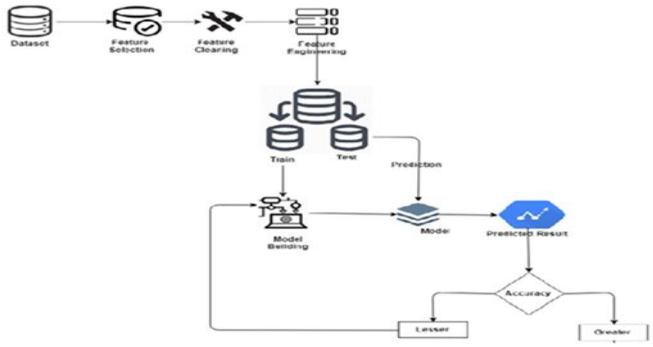
(2) We give the input video into a pre processor and break the genuine videos into n- number of images using frame separation . Each frame is about 30fps and the each frame has a image which is to be animated individually.

(3) We propose two basic yet compelling loss function works in GAN-based design. They are Generator and Discriminator

(3.1) In the generative organization, to adapt with generous style variety among videos and kid's shows, we present a semantic misfortune characterized as a $\ell 1$ meager regularization in the significant level element guides of the VGG network.

(4) Finally after all the above process we convert the frames of cartoonized images into a video stream and finally a cartoonized video is given as a output.

IV ARCHITECTURE



4.1Data collection:

To train an algorithm to detect sarcasm, we first need some data to train our algorithm on. Classification is a supervised learning exercise, which means we need to have some sentences labeled as sarcastic and sentences labeled as non-sarcastic so that our classifier can learn the difference between the two. In the next section we will discuss how to remove most of that noise, but short of reading all the tweets and labeling them by hand we cannot remove all the noise.

4.1.1Feature Selection:

Feature selection is based on which features will make an impact in our project and which feature we don't need to use. The Features we need to use are extracted from the dataset and other features are left as it is. The Feature can be multiple class as well as Single feature so we need to decide how our feature should come.

4.2.Datapreprocessing:

Before extracting features from our text data, it is important to clean it up. To remove the possibility of having sarcastic tweets in which the sarcasm is either in an attached link or in response to another tweet, we simply discard all tweets that have http addresses in them and all tweets that start with the @ symbol. Analysing of the data helps in screaming of the data carefully which can rectify misleading results. Pre-processing is done in three major steps like
i) Feature Extraction and Feature Engineering
ii) Feature Cleaning

1.Feature Extraction and Feature Engineering:

The text must be parsed to eliminate words, called tokenization. Then the words need to be determined as

integers or floating-point value for use as input to a machine learning algorithm, called feature extraction. Before applying feature extraction algorithms, the stemming of words was performed. Stemming is the process in which the words are shortened and normalized to their stem and their tenses are ignored. Several features are engineered so that might help the classification of tweets and I tested them on a cross-validation set.

2.Feature cleaning:

After the feature extraction we'll search for any null parameters in my data. If there are any null parameters there means we need to fill the parameters with the related content. In this process the steaming of the words are done. The similar words are being considered as one and the model is being built.

4.3 Training the system:

There is a very wide range of machine learning algorithms to choose from, most of which are available in the python library Scikit-learn. Precision is the number sarcastic tweets correctly identified divided by the total number of tweets classified as sarcastic, while recall is the number sarcastic tweets correctly identified divided by the total number of sarcastic tweets in the cross validation set. The F-score is simply the harmonic mean of precision and recall.

4.3.1 Model Selection list possible ways done by machine learning

i) Supervised learning

ii) Unsupervised learning

iii) Semi-supervised learning

iv) Reinforcement learning

Supervised learning:

Supervised learning is the task of inferring a function from labeled training data. By fitting to the labeled training set, we want to find the most optimal model parameters to predict unknown labels on other objects (test set).

Unsupervised learning:

In unsupervised learning we have less information about objects, in particular, the train set is unlabeled.

Semi-supervised learning:

Semi-supervised learning tasks include both problems we described earlier: they use labeled and unlabeled data.

Reinforcement learning:

RL is an area of machine learning concerned with how software agents ought to take actions in some environment to maximize some notion of cumulative reward.

4.3.2 CLASSIFICATION:

- Naive Bayes,
- Support Vector
- Machine,Random Forest

4.3.3 Measuring the model performance Confusion Metrices

True Positives (TP) –

These are the correctly predicted positive values which means that the value of the actual class is yes and the value of the predicted class is also yes.

True Negatives (TN)

These are the correctly predicted negative values which means that the value of the actual class is no and value of predicted class is also no.

False Positives (FP)

When actual class is no and predicted class is yes.

False Negatives (FN)

When actual class is yes but predicted class in no.

4.3.4 FEASIBILITY STUDY:

The objective of feasibility study is not only to solve the difficulty but also to obtain a sense of its scope. The key considerations are:

i) Economic feasibility

ii) Technical feasibility

iii) Social feasibility

V HARDWARE AND SOFTWARE SPECIFICATION

5.1 HARDWARE REQUIREMENTS

- Pentium Dual Core
- based processor at least 2.3 Ghz or higher
- Hard Disk -250 GB or Higher

5.2 SOFTWARE REQUIREMENTS

- Python(Pandas, Numpy,Sklearn)
- Anaconda, Jupyter Notebook and Spyder
- Flask 5000

- Python(Front end),NLP(Back end)

VI CONCLUSION

The way of up the existent caustic remark detection algorithms by as well as higher pre-processing and text mining techniques like emoji and slang detection area unit given. For classifying tweets as sarcasm and no sarcasm. There are various techniques used, however, the paper takes up a classification algorithm and suggests various improvements that directly contribute to the advance of accuracy. The Project derived analytical views from a social media dataset and also filtered out or reversed analyzed sarcastic tweets to achieve a comprehensive accuracy in the classification of the info that's given. The model has been tested in a time period and may capture live streaming tweets by filtering through hash tags so perform immediate classification.

VII REFERENCE

1. Wicana, SetraGenyang, TahaYasinİbisoglu, and UrazYavanoglu. "A Review on sarcasm detection from machine-learning perspective." 2017 IEEE 11th International Conference on Semantic Computing (ICSC). IEEE, 2017.

2. Selvan, LokmanyathilakGovindanSankar, and Teng-Sheng Moh. "A framework for fast-feedback opinion mining on Twitter data streams." 2015 International Conference on Collaboration Technologies and Systems (CTS). IEEE, 2015.

3. Suhaimin, MohdSuhairiMd, et al. "Natural language processing based features for sarcasm detection: An investigation using bilingual social media texts." 2017 8th International Conference on Information Technology (ICIT). IEEE, 2017.

4. Joshi, A., Tripathi, V., Patel, K., Bhattacharyya, P., & Carman, M. (2016). Are word embedding-based features useful for sarcasm detection?.arXiv preprint arXiv:1610.00883.

5. Dave, Anandkumar D., and Nikita P. Desai. "A comprehensive study of classification techniques for sarcasm detection on textual data." 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT). IEEE, 2016.

6. Hiai, Satoshi, and Kazutaka Shimada. "A sarcasm extraction method based on patterns of evaluation expressions." 2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI). IEEE, 2016.

7. Dave, A. D., & Desai, N. P. (2016, March). A comprehensive study of classification techniques for sarcasm detection on textual data. In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (pp. 1985-1991). IEEE.

8. Zhang, Meishan, Yue Zhang, and Guohong Fu. "Tweet sarcasm detection using deep neural network." Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers. 2016.

9. Bharti, Santosh Kumar, et al. "Sarcasm analysis on twitter data using machine learning approaches." Trends in Social Network Analysis. Springer, Cham, 2017. 51-76.

10. Ahmad, Tanvir, et al. "Satire detection from web documents using machine learning methods." 2014 International Conference on Soft Computing and Machine Intelligence. IEEE, 2014.

11. Dmitry Davidov, Oren Tsur and Ari Rappoport, Semi-Supervised Recognition of Sarcastic Sentences in Twitter and Amazon

12. Ellen Riloff, AshequlQadir, PrafullaSurve, Lalindra De Silva, Nathan Gilbert and Ruihong Huang, Sarcasm as Contrast between a Positive Sentiment and Negative Situation

13. Roberto Gonzalez-Ibanez, SmarandaMuresan and Nina Wacholder, Identifying Sarcasm in Twitter: A Closer Look

14. Christine Liebrecht, Florian Kunneman and Antal Van den Bosch, The perfect solution for detecting sarcasm in tweets #not

15. 2D to 3D conversion of dental images using deep neural network ,LM Leo, AJ Simla Journal of Chemical and Pharmaceutical Sciences 10 (3), 1432-1436 2 2017