End semester report on R & D Project (NU 302)

Academic Year- 2020-21

on

Sarcasm Detection Using Sentiment Analysis

A dissertation
Submitted in partial fulfillment of the requirements for the award of the degree
Bachelor of Technology

by

1. Somanshu Vikram Singh	**BT18GCS111**	**BTech-CSE**
2. Jatin Singal	**BT18GCS112**	**BTech-CSE**
3. Mayank Garg	**BT18GCS162**	**BTech-CSE**

Under supervision of
Dr. Ratna Sanyal

NIIT University, Neemrana, Rajasthan-301705

May 2021

NIIT UNIVERSITY
THE UNIVERSITY OF THE FUTURE
Industry-linked | Technology-based | Research-driven | Seamless

DECLARATION BY STUDENT(S)

We hereby declare that the project report entitled **Sarcasm Detection Using Sentiment Analysis** which is being submitted for the partial fulfilment of the Degree of Bachelor of Technology, at NIIT University, Neemrana, is an authentic record of our original work under the guidance of **Dr. Ratna Sanyal**. Due acknowledgements have been given in the project report to all other related work used. This has previously not formed the basis for the award of any degree, diploma, associate/fellowship or any other similar title or recognition in NIIT University or elsewhere..

Place: Neemrana

Date: 17 May, 2020

Somanshu Vikram Singh	**BT18GCS111**	**BTech-CSE**	Singh
Jatin Singal	**BT18GCS112**	**BTech-CSE**	Jatin
Mayank Garg	**BT18GCS162**	**BTech-CSE**	

NU
NIIT UNIVERSITY
THE UNIVERSITY OF THE FUTURE
Industry-linked | Technology-based | Research-driven | Seamless

CERTIFICATE BY SUPERVISOR(S)

This is to certify that the present R&D project entitled ……………………………………………….. being submitted to NIIT University, Neemrana, in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology, in the area of BT/CSE/ECE/GIS, embodies faithful record of original research carried out by ----------------------------------. She / He / They has / have worked under my/our guidance and supervision and that this work has not been submitted, in part or full, for any other degree or diploma of NIIT or any other University.

Place:

Name of the Supervisor(s) with signature

Date:

Acknowledgement

We are highly indebted to Dr. Ratna Sanyal for her guidance and constant supervision as well as for providing necessary information regarding the project and also for her support in completing the project.

Table of contents

Introduction **5**

Problem statement **7**

Literature review **8**

Proposed methodology **10**

Workflow 10

Technology 13

Result and Analysis **13**

Conclusion **13**

Future Scope **14**

References **15**

Annexure **15**

Introduction

Sentiment Analysis (opinion mining) is a text mining technique where we use Machine Learning and Natural language processing (NLP) to detect the sentiments/opinions of the overall text. These opinions/sentiments are crucial for further analysis in recognizing emotions, opinions, sarcasm, hate speech, criticism, and other such vital information. These sentiments can be mined from Social media platforms where users create their own content to convey sentiments about a topic, product or whatever they might like.

Sentiment analysis is a much broader field where identifying if the text is positive, negative or neutral is not enough. One might also need other crucial information like which emotions are dominant in the text, if the text is sarcastic, using emojis and gifs to make a statement, etc. Therefore, the output of text analysis should also include this other information too.

Sarcasm is itself of various types and differs from one individual to another. Some commonly known types are:

1. Self-deprecating sarcasm:

This type of sarcasm plays off of an exaggerated sense of worthlessness and inferiority.

EXAMPLE:

Boss : "Hey Bob, I'm gonna need you to work overtime this weekend."

Bob(employee): "Yeah, that's fine. I mean, I was gonna get married this weekend but, you know, it's not a big deal, I'll just skip it. She would've left me anyway."

This is a clear sarcasm where the employee tries to put himself down by saying that his wife would have let him if got married.

2. Brooding sarcasm:

The speaker says something polite and/or subservient in a bitter/irritated tone.

EXAMPLE:

Boss: "Hey Bob, I'm gonna need you to work overtime this weekend."

Bob: "Looking forward to it. I live to serve."

Here, the employee politely admits that he lives to serve, which is sarcastic in a subtle way.

3. Deadpan sarcasm:

This is a type of sarcasm said without laughter or emotion,so that it's hard to tell whether or not the speaker is joking with and/or mocking the other person.

EXAMPLE:

Boss: "Hey Bob, gonna need you to work overtime this weekend.".

Bob: "Can't make it. Got a cult meeting. It's my turn to kill the goat."

Here, Bob tries to evade by joking that he is in a cult and it is his turn to perform rituals. This way he can dodge overtime by not being aggressive.

4. Polite sarcasm:

This is a kind of sarcasm that sounds genuine at first, but then it slowly dawns on the listener that the speaker was just messing with the other person.

EXAMPLE:

Boss: "Hey Bob, I'm gonna need you to work overtime this weekend."

Bob: "Ooh, fun! I'll bring the ice cream!"

Bob doesn't deny overtime and rather answers in a polite manner to dodge the duty.

5. **Obnoxious sarcasm**:

It is the kind of sarcasm that is not really funny or clever, but it gets under your skin. It's usually spoken in a whiney tone of voice.

EXAMPLE:

Boss: "Hey Bob, gonna need you to work overtime."

Bob: "Oh, well that's just SO great. Just what I wanted to do this weekend. Awesome."

A sarcastic reply like this might irritate the other person.

6. Manic sarcasm:

It's a borderline crazy sarcastic comment/reply.

EXAMPLE:

Boss: “Hey Bob, I’m gonna need you to work overtime.”

Bob: “God, you are the best boss EVER! Have I ever told you how much I love this job? I wish I could live here! Somebody get me a tent, I never wanna leave!”

Such a reply from Bob will make the boss think he is a maniac, as no one usually loves their job to this extent or their boss. This can make the other person think that your reply was a sarcastic one.

7. Raging sarcasm:

This is more of a violent sarcasm with a hint of threat.

EXAMPLE:

Boss: “Bob. Overtime.”

Bob: “Oh, don’t worry! I’ll be there! Want me to shine your ******* shoes while I’m at it?! Hell, I’ll come to your house tonight and wash your goddamn Ferrari! Actually, you know what? Forget it. I’m just gonna go home and blow my brains out.”

As evident, Bob was violent with his reply with direct threats to hurt the person in front. The reply can be graphical in nature and not pleasant to hear in your day to day life.

Problem statement

Understanding a text is a difficult task because it depends on the perception of the entity. An entity needs to be defined that can process the data and classify it using various classification algorithms. There are multiple issues when it comes to sentiment analysis, namely, sarcasm detection, negation detection, word ambiguity, and multipolarity.

All these issues need to be dealt with, to make an accurate model, which can perform sentiment analysis on a given text. Detecting sarcasm in conversations is already difficult for many people, now to train a model which does the same job would be tough, but not impossible. However, the model’s accuracy would depend highly on the dataset we feed into the model. If the dataset is uniform with little to no variations in the sentiments portrayed by each text, then the model may fail to determine even an obvious text as sarcastic or not.

Apart from the dataset, sarcasm can be found in the single-line text, as well as hidden in reply to someone else's comments along with a sarcastic twist at the end of a paragraph that would otherwise be a non-sarcastic paragraph. These need to be classified as well as accounted for when training our model and preparing the dataset.

The final objective is to create a model that can detect sarcasm while taking into account all the issues mentioned above. Even if the accuracy falls shorter than the standard models, but it takes into account all the issues mentioned above, then further analysis would be required to improve the model.

Literature review

Text mining and sentiment analysis has been well researched and is in practice for quite some time. A recursive-SVM model for analysis and for data processing [1], where the authors have used Stanford constituency parser for parsing the tweets from the Twitter dataset. Further, they used RNN (Recurrent Neural Network) to store temporal information in the temporal memory component, among multiple implementations of RNN, LSTM (Long Short Term Memory) are easier to train as there are no exploding gradients. CNNs can also capture the same temporal sequence but they are limited when it comes to increasing word length and size of backpropagation. Adding a fully connected DNN on top of an LSTM network can help in mapping a better output. In the end, using a semantic model power of a neural network, f-score of 0.92 was achieved. However, further improvement can be achieved in identifying sentences similar at a conceptual level by improving the model by using word2vec.

A behavioural model approach to sarcasm detection in the Twitter dataset [2] has been done, where the user's past tweets are also taken into account when deciding whether a tweet posted by the user is sarcastic or not. This, SCUBA approach (as per Figure 1), provides us with a better understanding of data processing and feature extraction which can lead us to perform analysis on individual sets instead of a single big feature set. The python code is available [3] for SVM based classification using word2vec for further improving the model mentioned in [1].

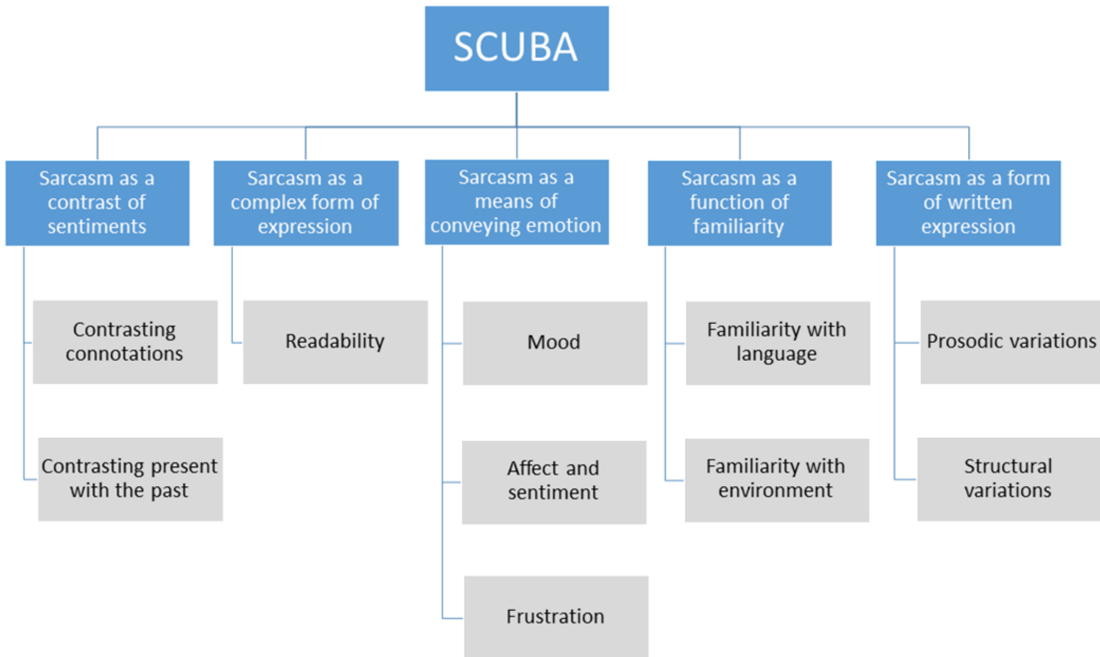

Figure 1. SCUBA framework image from [2]

In [4], the authors tried to identify the lexical and pragmatic factors that differentiate between sarcastic, positive and negative statements. They concluded that positive emotion is a significant feature in sarcastic and negative tweets but not in sarcastic and positive tweets while negation is a feature in positive and sarcastic tweets.

In another research work [5], the authors have used multiple approaches like Hash-tag based Approach, Contrast Approach, N-grams, Hyperbole Approach, Emoticon based, Hierarchical approach, Naive Bayes and finally Hybrid approach which is nothing more than a combination of all rules in one algorithm. A tweet is sarcastic if any one of the features is true except a Hashtag, as it deliberately tells that a tweet is sarcastic.

Literature also suggests that a methodology for identifying sarcasm on Twitter can be done using a Support Vector Machine (SVM) and Maximum Entropy Algorithms [6] with the following steps. First, they collected the data and created two data sets before adding the sarcastic tweets to training data and after adding sarcastic tweets to training data, Part-of-speech (POS) tagging was performed with Penn Treebank to mark each word with its associated speech portion. The authors extracted the characteristics such as punctuation, grammar, patterns, sentiment etc. from the training data. Once the characteristics have been extracted, the classification is carried out with SVM and Maximum Entropy Algorithms. And on the comparison, the Maximum Entropy Algorithm gives more accuracy than the SVM algorithm.

In a recent research work [7], the authors have performed a study on detecting sarcasm in Twitter and Amazon product reviews. They retrieved data by extracting #sarcasm on Twitter API. Data preprocessing is done in four steps as mentioned in Figure 2 and then data is passed to the Feature Extraction process which includes Term–Frequency, Feature Presence, Term Frequency–Inverse Document Frequency, Word2vec, etc. And then they apply data Classification algorithms like Random Forest, Gradient boost, Decision Tree, Simple Vector Machine (SVM), and Maximum Entropy to calculate the accuracy using precision and recall to identify the sarcasm on Twitter. And after comparing the results of each algorithm, high accuracy is given by the Maximum Entropy algorithm.

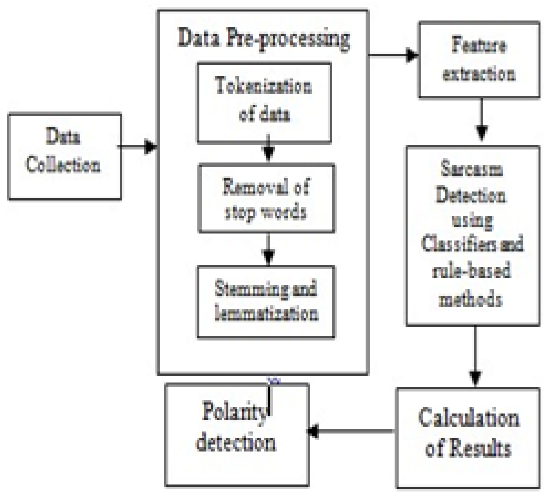

Figure 2. Sarcasm Detection Process [7]

Based on the related work survey, we are describing our proposed methodology to address the focused gap/challenge.

Proposed methodology

Workflow

In this paper we tried to implement various types of classifications like LSTM classifier (Figure 3) and Naive Bayes (Figure 4) classifier to determine whether an input text is sarcastic or not. Sarcasms are of different types as already mentioned in the introduction with examples. Every tweet is unique in its own way, it can be a comment to someone's tweet, a retweet, or indirectly hint at the unsaid rules in society. Therefore, accurately analysing tweets would be difficult. However, we will try to make use of the existing technologies to try and solve this problem.

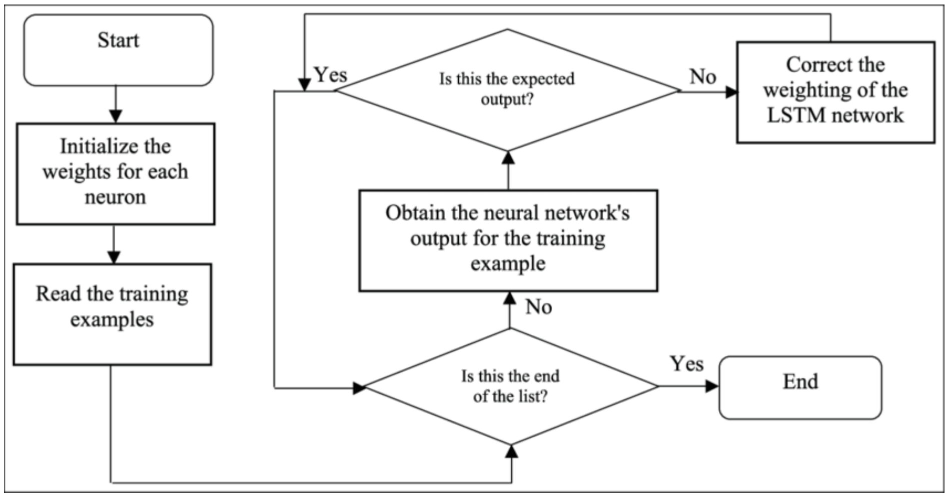

Figure 3. Flowchart for LSTM Classifier

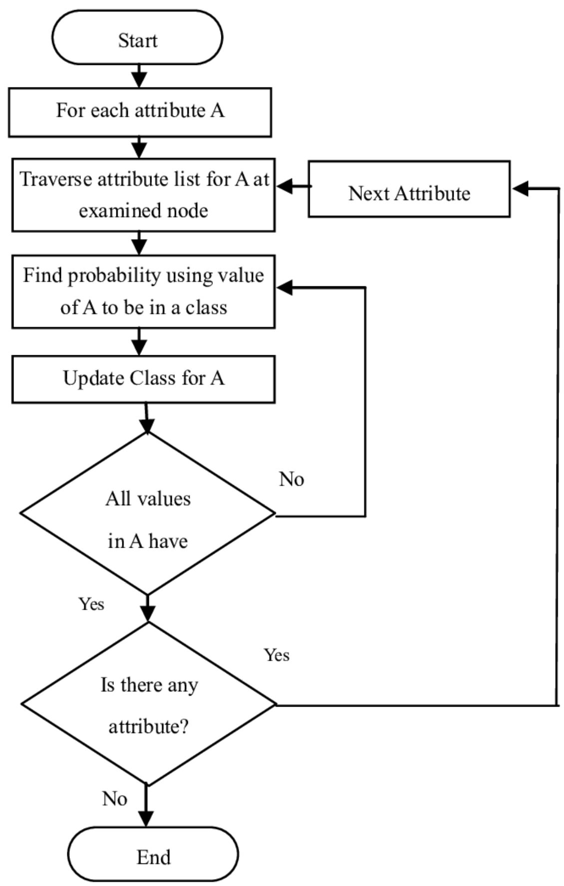

Figure 4. Flowchart for Naives Bayes Classifier

First, we scraped data from various data sites like Kaggle where label twitter data was available for academic purposes. The dataset's size is somewhere around 1.5million labelled tweets, stored in a CSV format. This data needs to be processed through a cleaning method to remove URLs (https://twitter.com), Retweets(RT), User Mentions(@*anonymous*). We also need to remove emoticons like(xD, :P), emojis, slangs and hashtags(#*not,* #*sarcasm*) but sometimes tweets are made sarcastic by the use of the emoticons , emojis, slangs and hashtags. Therefore we have to be careful while removing them from tweets.

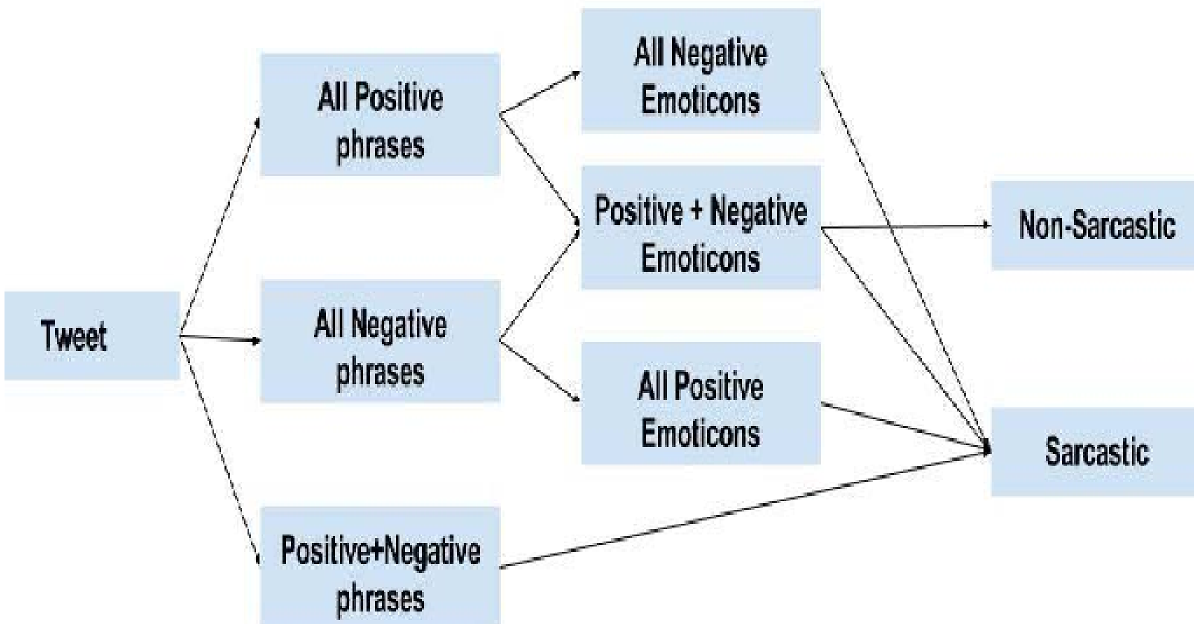

Figure 5. Design Implementation Diagram

We then created a filter for filtering out emoticons, emojis and slangs by replacing it with their implications directly in the tweet, therefore the classifiers can now understand them just like any other word. For example, a smiley emoji implies smile and positivity, therefore we replace the emoji with the word smiley, which is understood by parser.

The data is now cleaned and ready to be fed into the model. The data will be first tokenized to break it into smaller texts/subtexts and then split into the training and testing data. The training data will be used to train the model and it usually contains more data then the testing dataset as the model learns better with a high ratio of training data : testing data.

We have two types of classifiers for training the model, one is LSTM and the other is Naive bayes classifier. We will train both the classifiers using the cleaned dataset, and measure the accuracy of both these classifiers and compare them, the one with the best accuracy will be used for detecting sarcasm in unlabelled data.

Technology

For our project, we had opted Python as our programming language as it provides a lot of machine learning modules and libraries. In our work, we had used Tensorflow along with Keras and sk-learn.

Currently, we have implemented two different classifiers, LSTM and Naive-bayes. LSTM is done with the help of Keras, where a model has been trained on news headlines and saved for our testing purpose, while another model with over 1.5 million tweets has been created. For Naive Bayes, we were only able to train it on the Headline dataset. Since both models have been trained on the News Headline dataset, it provides us a comparison ground between them.

We trained our models on Google Colab, instead of our physical devices, since Google colab provides ample memory to load the 1.5million tweets for processing and also google colab allowed us to share the source code amongst our team members.

Result and Analysis

We were able to get an accuracy score of 0.81+, which is very good, considering the industry average to be around 0.70. The model is able to always correctly identify if the data is sarcastic or not, if it is picked from the 1.5 million tweets dataset which we used for training. However, when working on new unlabelled data, it seems to identify the tweet correctly 8/10 times, but it also sometimes incorrectly labels the data, sarcastic when data is non-sarcastic and non-sarcastic when data is sarcastic. This can be rectified further in future scope of this project.

With this we analyzed that LSTM, an artificial recurrent neural network, performed better than Naive Bayes, as a RNN keeps a “memory” of what it read previously and so it is also better option for contextual sarcasm, where the sarcasm is identified only if there is a prior knowledge/memory to build up on.

Conclusion

Because of the encouraging results obtained, future work could definitely be pursued in the direction of expanding the feature set in order to include more features that are expressive of the user's behavior. Moreover, the detection of sarcasm in tweets is limited, in the sense that the sentences are of limited sizes. The same approach could be expanded to detect sarcasm on other social media platforms or product reviews. This could go a long way in restricting the spread of fake news on account of people not recognizing the posts as sarcastic. On an ending note, the current approach works on a static dataset. The possibility of adding incremental classification capabilities could also be attempted in the future.

Future Scope

In our project, we faced a lot of challenges like which model to use, is the dataset size good enough and many more. Now, we know few ways by which we can improve the accuracy of our program

1. Using a much larger dataset: Dataset is crucial for any model, the larger the dataset, the better the model. Currently, we have trained our model only on Twitter dataset and News headlines. For a better model, the dataset must be trained on a combination of news headlines, twitter tweets, chats and conversation. After that, the model needs to be tested on various different dataset such as conversation and reviews.
2. More training time: We all know that training models requires too much computational power and as the dataset size increases, the time to train the model increases. Apart from this, selecting the batch size and epochs are challenging. Increase the epoch size, the model may be overfit on the data. Keep the batch size small, the time to train the model increases exponentially. Therefore, selecting a specific training parameter is needed.
3. SCUBA method can be implemented: We tried to implement the SCUBA framework but we found out that this framework is too theoretical and complex for us to implement. But we believe that SCUBA holds great potential to generate better results when combined with other models.
4. Hybrid approach seems feasible. Hybrid approach is nothing more than a combination of all rules in one algorithm. Currently, we were able to implement the basis of hybrid approach by implementing two different algorithms for our comparison. For us to make sure that hybrid approach works perfectly, we need to implement multiple algorithms and train the model on the output of their results to receive better and reliable results.

References

[1] Aniruddha G. and Tony V., “Fracking Sarcasm Using Neural Network”, 7th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis, At San Diego, California, USA, Volume: pp 161–169, 2016.

[2] Ashwin R., Reza Z., and Huan L., “Sarcasm Detection on Twitter: A Behavioral Modeling Approach”, Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, ACM, pp 97-106, 2015

[3] Chinmay Jain (2015) “Sarcasm Detection in Twitter Data using SVM Classification”, https://github.com/chinmayjain18/Sarcasm-Detection-in-Twitter-Data-using-SVM-Classification .

[4] Roberto G., Smaranda M. and Nina W., “Identifying Sarcasm in Twitter: A Closer Look”, Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers- Volume 2, pp 581—586, 2011

[5] Rathan K and Suchithra R, “Sarcasm detection using combinational Logic and Naïve Bayes Algorithm” Imperial Journal of Interdisciplinary Research (IJIR) Vol-3, Issue-5: pp 546-551, 2017.

[6] Sana P, Sachin N. D, “Opinion Mining in Twitter – Sarcasm Detection” International Research Journal of Engineering and Technology (IRJET), volume 04, issue 10, pp 201-204, 2017.

[7] Sindhu. C, G. Vaidhu, Mandala V R, “A Comprehensive Study on Sarcasm Detection Techniques in Sentiment Analysis”, International Journal of Pure and Applied Mathematics, volume 118, pp 433-442, 2018.

Annexure

Source code is too big to attach in document, source code will be shown during live demonstration.