2021 2nd Global Conference for Advancement in Technology (GCAT)
Bangalore, India. Oct 1-3, 2021

Identification of Hate Speech using Natural Language Processing and Machine Learning

Darsh Bhimani
Computer Engineering
Shah and Anchor Kutchhi Engineering College
Mumbai, India
darsh.bhimani@sakec.ac.in

Rutvi Bheda
Computer Engineering
Shah and Anchor Kutchhi Engineering College
Mumbai, India
rutvi.bheda@sakec.ac.in

Femin Dharamshi
Computer Engineering
Shah and Anchor Kutchhi Engineering College
Mumbai, India
femin.dharamshi@sakec.ac.in

Deepti Nikumbh
Assistant Professor
Shah and Anchor Kutchhi Engineering College
Mumbai, India
deepti.nikumbh@sakec.ac.in

Priyanka Abhyankar
Assistant Professor
Shah and Anchor Kutchhi Engineering College
Mumbai, India
priyanka.abhyankar@sakec.ac.in

2021 2nd Global Conference for Advancement in Technology (GCAT) | 978-1-6654-1836-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GCAT52182.2021.9587652

***Abstract*— From the past decade, social media has gained a lot of momentum both in a positive way as well as in a negative way. With this rapid increase of networking through social platforms and websites, people are able to communicate with each other directly with no cultural or economic gap. While there have been many benefits of social media but there are no less negative impacts on the society. One such problem that has arised since the past few years is the hate speech. Hate speech is basically the use of offensive and hostile language happening on the social media. It may refer to any individual or a certain group of people with the same interests. In this paper, we have introduced our way of dealing with this hate speech and minimizing it to a large extent. People convey their hatred and anger straightaway on social media which would hurt the feelings of other people. It would affect their caste , creed, religion, race and would have a very negative impact on them. Some comments might not be intentional to anyone but would be counted as hate speech due the foul language used. We have dived deep into natural language processing to eliminate hate speech and used various machine learning models to decide which one to use as per its accuracy.**

Keywords—Hate Speech , Machine Learning, Social Media, Natural Language Processing

I. INTRODUCTION

Over the past few years, social media has been the major medium of communication throughout the world to convey ideas. As per reference [1], social media is actually a trustful platform but since there is excess amount of information posted and discussed, it becomes a little difficult to scan a comment in single resulting in increase of hate speech. With this rapid increase of networking through social platforms and websites, people are able to communicate with each other directly with no cultural or economic gap. Hate speech can also be explained as a concept of emotion. We can understand hate speech as the use of offensive and hostile language happening on the social media. It may refer to any individual or a certain group of people with the same interests. In this paper, we have introduced our way of dealing with this hate speech and minimizing it to a large extent.

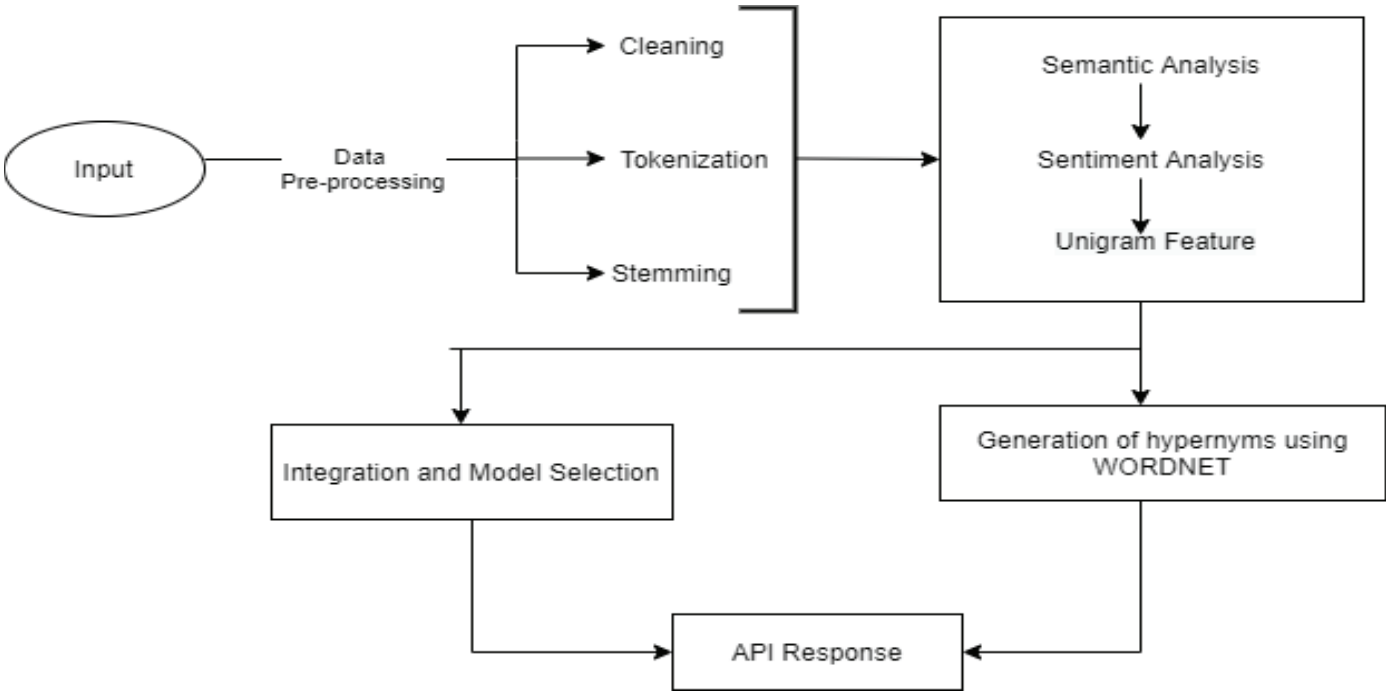
Hate speech has increased a lot since the past few years. Infact it has furthermore increased ever since the lockdown has occurred due to the COVID-19 pandemic because all work and communication has been online ever since then. Social media platforms like Twitter, Facebook, and Instagram are becoming increasingly common among people of all ages, cultures, and interests. These are some of the platforms where hate speech has been occuring. These platforms provide an open stage for people to voice their opinions and share or convey their thoughts and messages across the world but the enormous number of posts and exchange of messages makes it almost impossible to keep their content in control. Facebook has a set of community guidelines[2] in place to deal with abuse, online bullying and illegal behavior, sexual assault and violence on public figures. Similar to Facebook, Twitter too has some guidelines[3] that would aid someone who is a victim to social abuse.

II. RELATED WORK

Hate speech does not only creates chaos and tension between various groups but also leads to real-life problems[4]. This paper explains what is hate speech and gives various examples as to how hate speech actually occurs. It primarily focuses on Twitter for dealing with hate speech. Next it shows the data pre-processing. It further explains the methods that are used on the dataset like the Sentiment analysis, Semantic analysis, Unigram feature followed by the Pattern extraction. It has different graphs and tables to show the precision and accuracy achieved for different models.

Reference[5] explains about pronunciation features to detect cyber-bullying. The pronunciation features can help in the identification of misspelled and censored words. It starts with the tweet pre-processing followed by feature extraction and then word embeddings. It explains the frequency based and prediction based word embeddings.

Fig. 1. System Architecture



Next, a combination of lexicon based and machine learning approaches[6] to predict hate speech contained in a text. This paper gives examples of the most frequently used words in any comment or paragraph and states how often it has been used. The words show us the emotion like be it sadness, anger joy and so on. This is a very helpful input. This paper hence uses an emotional approach through sentiment analysis.

LSTM is yet another method to detect hate speech[7]. It explains LSTM along with the components used in it. Various tables have been shown presenting the recall, precision of the result. An epoch test comparison chart is generated to compare various accuracies. Lastly, we have a system that constructs word embeddings called as the Word2vec[8]. It performs a qualitative evaluation of the most similar words to a few target words. It also highlights that the similarity by semantic analysis does not necessarily mean synonymy.

III. Proposed Approach

We studied various research papers and dived deep into various machine learning models. We also dived into Natural Language Processing and its various features and methods that will be used for this project. Firstly, we divided our tweets into two classes: Clean and Hateful. The first class: Clean, as the name suggests would include all the data or comments that are neither offensive nor hateful. These tweets are unbiased and do not involve hate speech in them. The second class is Hateful which would consist of the comments containing hate speech and might be offensive to particular segment of people. It would consist of hateful, racist and offensive comments. Clean or Hateful, these are the end result that we will be getting after processing a particular comment.

Fig 1. explains the system architecture that we aim to follow for the complete system. Initially, an input will be given followed by the data pre-processing which is further consists of the cleaning of the comment, tokenization and stemming. After pre-processing, the natural language processing methods will be used i.e. the semantic analysis, sentiment analysis and unigram extraction. This will be followed by the generation of words with similar context with the help of WORDNET. On the other side the integration and selection of the model will take place. And lastly, an API response will be sent and an output will be produced. All these methods and steps are explained further in the paper.

The proposed approach to be used is as follows.

A. Data

The dataset used is the Automated Hate Speech Detection and the Problem of Offensive Language[9] which consists of around twenty five thousand tweets. The labels used are hate speech, offensive language and neither. Neither set consists of a positive sentiment content which does not belong to any of the other two classes. The next dataset used is the Hate Speech Dataset from a White Supremacy Forum[10] consists of around two thousands comments. This dataset is of a different distribution compared to the first one. The labels in this dataset are hate and no hate. These are the datasets used on which data pre-processing will be done next.

B. Data Pre-processing

- Cleaning: A comment might contain various tags which should be removed before using or performing any operations on it. A tag is addressed with an @ which would direct the specific comment to a particular person or an organization. Presence of a tag is not useful since it does not exactly tell us the content of that data. Besides the tags , a comment may even contain various URLs or links to other webpages. URLs would re-direct us to some other webpage which might not be useful to us as our main criteria is to eliminate hate speech present on comments of various social platforms. Hence, these too should be removed from the comment. Hashtags is another object that should be removed from the comment. The word with the hashtag would be of help to us as it would be

related to the content itself hence hashtags should be eliminated from the comment.

- Tokenization: The next step would be tokenization. Tokenization basically means splitting up of the given comment or phrases into words. Large chunks of text is broken into smaller parts for better use of it.
- Stemming: Stemming would bring the word to its base form without changing its meaning. Say for example if we have the word running , so this word can brought to run by stemming. Porter Stemmer is one of the NLTK library that will be used to perform this step.

C. Methodology

- **Semantic Analysis:** This analysis explains the grammar part of the comment. It included how the user uses exclamation marks, question marks and other such punctuations. Sometimes in a comment , the use of punctuation can actually let us know the real meaning behind that comment. A comment with and without any punctuation can make a lot of difference. Say for example : "You better stop it now!" . Here the exclamation mark makes us understand that this comment might act as a warning for someone. Had it not been there then it would simply mean a normal sentence with neutral meaning. Hence this is a very important feature as it helps us to know the meaning and context of the comment posted at the first place with the help of the capitalization and other punctuations.
- **Sentiment Analysis:** As the word suggests, sentiment, means how the content is affecting others interests and sentiments. It can be a person or a particular group of people. We use sentistrength here which enables us to know the strength of the sentiment be it too strong or neutral or light. This feature majorly helps us to know the content whether it is positive or negative. However care should be taken as text categorization is often inaccurate due to sarcasm and subtexts[11]. However it is not necessary that every negative comment is hate speech and hence we need another method to further detect hate speech. It consists of positive and negative score.
- **Unigram Feature:** This feature consists of phrases that are specifically hateful. In this the hateful words that are used more often are known and a column vector is formed. The words that we get are later taken into consideration as features. This would get us more nearer in knowing if it is a hate speech or not.

IV. Experimental Setup

We implemented our combined output of dataset performed with all operations for an auto model run. Auto Model of Rapid Miner basically helps in enhancing the process of building models. They address issues like prediction, clustering and outliers. Prediction helps in solving classification as well as regression problems. It also helps in comparing the outputs of these models when the calculations are over. The table I shows the results that we got.

TABLE I. Performance Measures Of Different Models

Method	Precision	Recall	F measure	Accuracy
Naïve Bayes	0.511	0.100	0.677	0.511
Generalized Linear Model	0.977	0.951	0.964	0.963
Logistic Regression	0.983	0.946	0.964	**0.964**
Decision Tree	0.990	0.929	0.958	0.958
Random Forest	0.983	0.836	0.904	0.909
Gradient Boosted Trees	0.988	0.937	0.962	0.962
Support Vector Machines	0.981	0.942	0.961	0.962

Table I shows the performance measures of the different machine learning models by auto model run. The different parameters considered here are the precision, recall, F-measure and the accuracy. We got the maximum accuracy for the logistic regression model. The accuracy achieved was 0.964. This was one of our experimental setup to check how the models work on this platform.

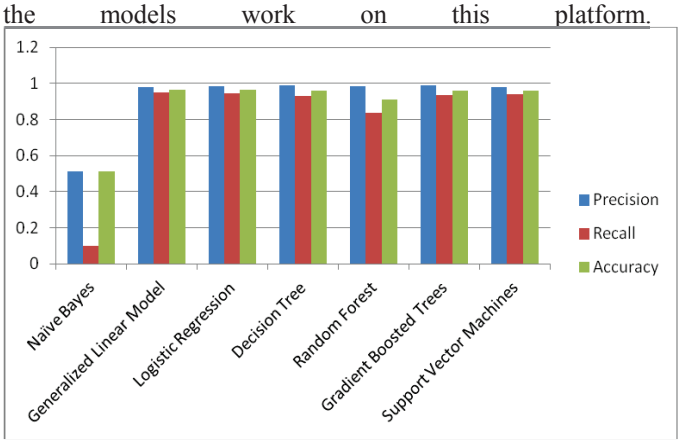


Fig. 2. Bar graph of all the parameters

Fig 2 depicts the graph form of the above table given. The parameters considered in this graph are precision, reacll and accuracy. The x-axis contains all the machine learning models used and the y-axis consists the values that have been achieved. It gives us a better understanding as to what method show what value of a specific parameter just at a glance. With this, we complete our first experimental setup of the auto-model execution.

With the above proposed approach and the classes formed, we tried several machine learning models and classifiers to check which one would provide us the maximum accuracy. After running in the auto model, we tested the same dataset on a Python IDE. The same dataset was executed on it to see the output. Table II shows the accuracies obtained with the respective model.

TABLE II. ACCURACY ACHIVED ON A PYTHON IDE

Classifier	Accuracy achieved
Ada Boost Classifier	0.9561
Decision Tree Classifier	0.9467
Gaussian NB	0.9454
Gradient Boosting Classifier	0.9494
Linear Discriminant Analysis	0.9451
Logistic Regression	**0.9629**
Random Forest Classifier	0.9550
Quadratic Discriminant Analysis	0.8706

Table II shows us the accuracy achieved when the above machine learning models are executed on a Python IDE. We get the maximum accuracy of 0.9629 for the logistic regression model. Hence we have executed through two platforms and achieved various parameters and results on different machine learning models.

V. CONCLUSION & FUTURE WORK

Hate speech is very harmful and hence needs to be dealt with very seriously for its elimination. It does not only lead to real life conflicts but also affects the mental health of a person. Some people cannot take the hate speech and it affects them quite badly. In this paper, we have stated our way to detect any type of hate speech. We briefly explained the various features of Natural Language Processing that were used. We implemented various machine learning classifiers and selected the one with the maximum accuracy. One of the use case for the future work in this system could be the detection of long phrases or sentences. Another use case can be the detection of a person's behavior from its past posts and comments and hence limiting the spread of hate speech.

ACKNOWLEDGMENT

Special thanks to our guide Prof. Deepti Nikumbh, Assistant Professor, SAKEC and co-guide Prof. Priyanka Abhyankar, Assistant Professor, SAKEC for giving their insightful inputs and guiding us throughout this project.

REFERENCES

[1] Graeme Foux. Consumer-generated media: Get your customers involved. Brand Strategy, 8(202):38–39, 2006.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. ,1892, pp.68–73.

[2] https://www.facebook.com/communitystandards/

[3] https://help.twitter.com/en/forms/safety-and-sensitive-content/abuse

[4] M. Bouazizi, H. Watanabe and T. Ohtsuki, "Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection," in IEEE Access, vol. 6, pp. 13825-13835, 2018, doi: 10.1109/ACCESS.2018.2806394.

[5] A. Shekhar and M. Venkatesan, "A Bag-of-Phonetic-Codes Modelfor Cyber-Bullying Detection in Twitter," 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT), 2018, pp. 1-7, doi: 10.1109/ICCTCT.2018.8550938.

[6] M. Gomes, R. Martins, J. J. Almeida, P. Henriques and P. Novais, "Hate Speech Classification in Social Media Using Emotional Analysis," 2018 7th Brazilian Conference on Intelligent Systems (BRACIS), 2018, pp. 61-66, doi: 10.1109/BRACIS.2018.00019.

[7] S. S. Syam, B. Irawan and C. Setianingsih, "Hate Speech Detection on Twitter Using Long Short-Term Memory (LSTM) Method," 2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), 2019, pp. 305-310, doi: 10.1109/ICITISEE48480.2019.9003992.

[8] Mohammed, Nora. (2019). Extracting Word Synonyms from Text using Neural Approaches. The International Arab Journal of Information Technology. 45-51. 10.34028/iajit/17/1/6.

[9] Davidson, Thomas & Warmsley, Dana & Macy, Michael & Weber, Ingmar. (2017). Automated Hate Speech Detection and the Problem of Offensive Language.

[10] de Gibert Bonet, Ona & Perez Miguel, Naiara & García-Pablos, Aitor & Cuadros, Montse. (2018). Hate Speech Dataset from a White Supremacy Forum. 11-20. 10.18653/v1/W18-5102.

[11] Niraj Pal, Dhiraj Gurkhe, Rishit Bhatia, "Effective Sentiment Analysis of Social Media Datasets using Naive Bayesian Classification", International Journal of Computer Applications (0975-8887), Volume 99 - No. 13, August 2014.