Twitter HateSpeech Classifier Using NLP

Alexender Kissiedu
IT Training and Support Section
Directorate of ICT Services
Cape Coast, Ghana
PS/MCS/21/0007

Evans Ankomah
IT Training and Support Section
Directorate of ICT Services
Cape Coast, Ghana
PS/MCS/21/0009

***Abstract*—The main of this paper was to create a method of identifying hate word on twitter using machine learning (ML) algorithms. We tried creating models using Logistic Regression and Support Vector Machine (SVM) ML algorithms in combination with frequency–inverse document frequency (Tf-idf) and Bag of Words(BoW) features extraction models. At the end the final model was SVM with hyper tunned BoW feature extraction model. This resulted in an accuracy score of 95.14%.**

Keywords—NLP, HateSpeech. Logistic Regression

I. Introduction

Currently, we have the freedom in our current world to share our opinions, beliefs, criticize others, and comment on others' opinions on various social media platforms. This practice has been made feasible by the developing democracy in different countries' as well as the use of social media networks and the sharing of people's thoughts through them. The worst part is that some people even believe that using hate words against other people on social media platforms is natural, but in fact doing so is not courteous or polite to those who receive these remarks or who just see these words with improper vocabulary while skimming through other tweets.

Hate speech is defined as hurtful or demeaning discourse that expresses preference towards a particular group, often on the basis of race, religion, or sexual orientation(Gitari et al., 2015). So it has now become very significant to control or identify hateful speech. All democracies believe that hate speech must be banned if it calls for the harm or violence of one group against another. Regulating hate speech that is derogatory of particular groups of people has not been possible for Americans who uphold the fundamental right to free speech guaranteed by their Constitution, and it cannot be done now. According (*Eggheads Have Found a Positive Link between the Number of Racist Tweets and the Number of Racist Hate Crimes in US Cities • The Register*, n.d.) study, there is a direct correlation between the number of racist tweets and actual hate crimes in 100 US locations. Twitter has been under fire for failing to take action to stop hate speech on its platform. Since criminalizing such speech would go against the 1st Amendment's protection of the rights granted to its citizens, democracies around the world adopt much tougher measures to suppress it than the United States does.

Recent occurrences, have shown how words have the potential to drive people to violence. We saw how the surge in Asian-American hate crime(*Addressing Hate Crimes Against Asian Americans and Pacific Islanders*, n.d.) resulted in the Atlanta shooting as a result of constant usage of words like "Kung-flu" or "Chinavirus" that link COVID to China.

This paper therefore aims to detect and classify hatespeech on Twitter using Tf-idf and BoW models coupled with LR and SVM ML algoritthms. The jupyter notebook of this paper include the process of data cleaning, data preprocing, visualization, feature extraction and the model building. These models were build using label tweets form (*Twitter Sentiment Analysis | Kaggle*, n.d.). The models were trained differentiate between hateSpeech and non hateSpeech and then analyze the results in order to better understand the data. At the end our model was able to predict 95.14% correctly.

II. Related Works

The assessment of subjective speech on social media platforms has been thoroughly researched and applied in various domains, including sentiment analysis(Babu & Kanaga, 2022; Balli et al., 2022; Hodeghatta, 2013), sarcasm detection(Venkatesh & Vishwas, 2021) and gossip detection(Lin et al., 2021), among others. Several methods and approaches can be used for detecting of hate speech. (Warner & Hirschberg, 2012) specifically targeted online sentences.

Despite several approaches in achieving this, automatic detection of hate speech on social media still remains a bit challenging due to the separation between offensive and hate language. (Davidson et al., 2017) made an initial twitter collection utilizing hate speech keywords. The tweets were divided into the following three categories using crowdsourcing: "hate speech," "offensive language," and "neither." The three categories were then separated using a multi-class classifier that was trained. 90% of the F1-Score were earned by the model that performed the best. The confusion matrix, however, showed that approximately 40% of the tweets containing hate speech were misclassified.'

However, it is very significant to develop more robust and accurate models for detecting hate speech. It is also very important to build model using a combination of several ML algorithms in combination with different feature extraction models to determine which one will give the best score.

III. Dataset description

The Dataset used in this paper was obtained from (*Twitter Sentiment Analysis | Kaggle*, n.d.)and can be easily downloaded from their website. The repository provides two datasets. It has train.csv for training the model and the test.csv which contains 17197 tweets for testing the model. Table I shows the distribution of the training datasets.

TABLE I. Classwise Distribution of training dataset

Class	***No. of instances Subhead***
Class 0	29720
Class 1	2242

[a] Class 0 – Non -Hate Speech

[b] Class 1 – Hate Speech

IV. DAta preprocessing

There are several ways used by machine learning models to achieve higher evaluation. Below are the methods used in this work.

A. Tokenizing

In this stage each phrase or sentence is broken down into individual words during the tokenizing process. For our dataset, this is utilized to generate a vocabulary. And each vocabulary is used to identify unique tweets in our dataset, and its use depends on the approach we choose, ether Tf-idf or BoW approach.

B. Stop Words

Stop words are those words that have little or no meaning, such as a, the, of which are frequently used in most phrases. Therefore, removing these stop words is necessary to avoid misclassification. So as part of our preprocessing we did remove all the stop words before building are model to avoid misclassification

C. Lemmetization

Lemmatization takes context into account and changes the term to its logical base form, called Lemma. For instance, lemmatizing the word 'Caring' would return 'Care'. We could have also used stemming at this stage but stemming unlike lemmatization might not always lead to providing a logical or dictionary word. The choice here absolutely depends on the dataset and the model you are building. But like stated already we decided to go with lemmetization, hence we applied lemmatization to the tweets.

D. Others

Similarly, we performed other preprocessing such as casefolding – coverting all text to lowercase. We also removed retweet and quote tags, URLs, white spaces, numeric numbers and words with digits, punctuations and trailing white spaces from our dataset.

V. Feature extraction

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. This leads to better results than applying machine learning directly to the raw data. It is used identify key features in the tweet data for coding and by learning from the coding of the original data set to derive new ones. This is one of the key preprocessing techniques used in mining data and classifying text that measures the context of documents(Wen & Zhang, 2017). For the purpose of our work, we used Tf-idf, BoW model and N-grams.

A. Tf-idf

This is the most recognized and widely used as a balancing technique and its effectiveness is equivalent to modern techniques. Documents are known to be variables in the word weighting. The key preprocessing method required for indexing text is selecting a function for a function selection procedure(Ramya & Pinakas, 2014). This is done to determine how frequently a word appears in a manuscript. The Tf-idf calculates the term frequency first, then uses it to determine the inverse document frequency. Finally, it multiplies and normalizes the two numbers above to provide an aggregate.

B. Bag of Words (BoW)

We also used the BoW model to evaluate textual data for the ML algorithm and allows us to achieve our goal of predicting hate speech.The BoW approach results in an illogical representation of the document. For example the sentence "John likes to watch football. Mary likes football too", the BoW concept will not reveal that the verb "likes" always comes after a name of a person in this text. BoW model can be seen a special case of N-grams model.

C. N-grams

An N-gram is a sequence of N tokens (or words. It comprises of unigram, bigram and trigrams. A unigram is a one-word sequence. for instance, performing a Unigram on the sentence "I love to data Machine Learning Related post" will be: "I", "love", "to", "data", "Machine", "Learning", "Related", "post". Similarly, a bigram will result in words like: "I love" , "Machine Learning", "Related post" . and lastly a trigram will result in words like: "I love to". "data Machine Learning", "Learning Related post".

At any point an N-gram language model will predict the probability of a given N-gram within any sequence of words in the language. However, increasing the N will result in lots of computation overhead that will requires large computation power in terms of RAM. Nonetheless, the higher the N, the better it is for the model. In this work however we resulted to using trigrams for our models

VI. Visualization

Figure 1 demonstrate the how the tweets are distributed among hate speech (1) and none hate speech (0). We saw the only about 6.5% of the tweets contain hate speech. While the remaining 93.5% contain non hate speech. In figure 2 also we see a visualization of the most frequent words (such as love, day, time, happy and today). In figure 3 also we see words like Trump, white, black, racist, allahsoil coming up strongly as hate tweets.

Fig. 1. Distribution of Hate Speech against Non hate speech.

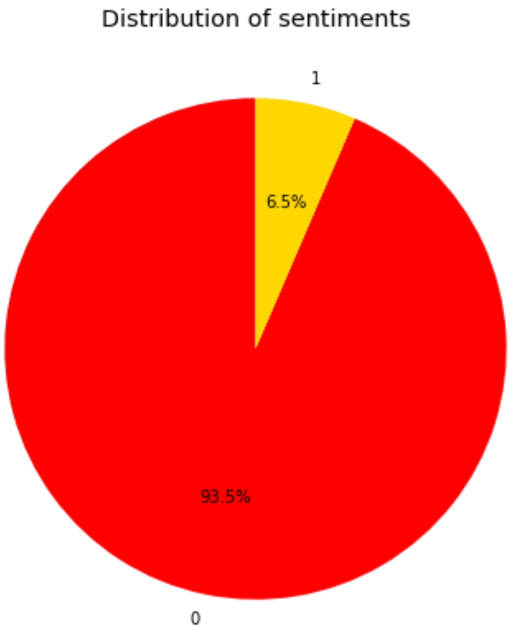


Fig. 2. Most frequent words in non hate speech.

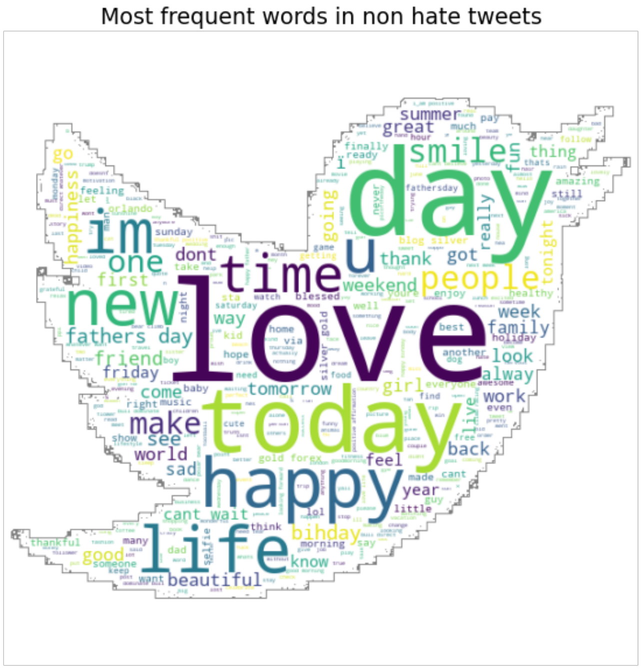
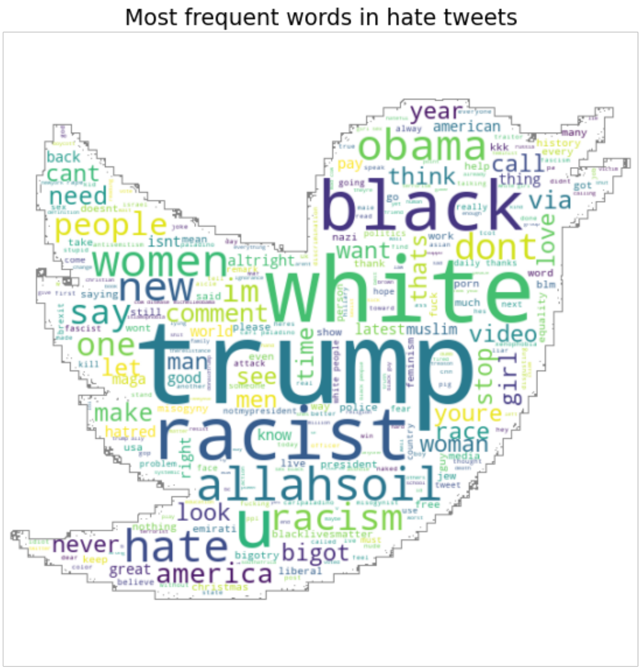


Fig. 3. Most frequent words in hate speech



VII. Results

Below are the results of the confusion metrices after performing Logistic Regression and SVM algorithms on the tweets with tf-idf and BOW representation. Table II Shows the matrix prior to the hyper parameter tunning and table III shows the matrix after the hyper parameter tunning. Table Iv also displays the accuracy score both models before and after hyper tunning.

Comparing the accuracy scores, we released that our model achieved slightly higher score in all cases after performing the hyper parameter tunning on the dataset. For our hyper tunned Logistic regression model, we achieved an accuracy score of 94.54% with the Tfidf approach while we recorded 95.09% with BoW. Similarly, the support vector machine algorithm gave us an accuracy score of 94.96% with the Tfidf approach as against 95.14% with BoW.

TABLE II. CONFUSION MATRIX BEFORE HYPER PARAMETER TUNING

	LR with Tfidf	
	Non-Hate Speech	Hate Speech
Non-Hate Speech	5261	2
Hate Speech	338	6
	SVM with Tfidf	
Non-Hate Speech	5258	5
Hate Speech	304	90
	LR with BoW	
Non-Hate Speech	5227	36
Hate Speech	226	128
	SVM with BoW	
Non-Hate Speech	5191	72
Hate Speech	209	185

TABLE III. CONFUSION MATRIX AFTER HYPER PARAMETER TUNING

LR with Tfidf		
	Non-Hate Speech	Hate Speech
Non-Hate Speech	5257	1
Hate Speech	303	91
SVM with Tfidf		
Non-Hate Speech	5255	8
Hate Speech	277	117
LR with BoW		
Non-Hate Speech	5211	52
Hate Speech	226	168
SVM with BoW		
Non-Hate Speech	5237	26
Hate Speech	249	145

TABLE IV. ACCURACY SCORE

Model	Before Hyper Tuning	After Hyper Tuning
LR with Tfidf	93.11%	94.54%
SVM with Tfidf	94.54%	94.96%
LR with BoW	94.66%	95.09%
SVM with BoW	95.03%	95.14%

VIII. CONCLUSION

From out work classification model using tf-idf and bag of words methods to extract feature from the tweets. and applying the Logistic Regression and SVM machine learning algorithms. We can conclude from our results that using hyper tuned SVM with BoW gives that best performance with an accuracy score of 95.14%.

It is important to note also that even with or without gridsearch hyper parameter tunning, SVM with BoW still gives the best score. This is so because instead of using ratio of term frequency to the document frequency in the case of Tf-idf, BoW just count the frequency of words and uses it as a vector.

However, more studies is needed in classifying hatespeech especially by applying several ML algorithms in combination with different feature extraction models.

REFERENCES

Addressing Hate Crimes Against Asian Americans and Pacific Islanders. (n.d.). Retrieved December 2, 2022, from https://www.justice.gov/hatecrimes/addressing-hate-crimes-against-AAPI

Babu, N. V., & Kanaga, E. (2022). Sentiment analysis in social media data for depression detection using artificial intelligence: A review. *SN Computer Science*, *3*(1), 1–20.

Balli, C., Guzel, M. S., Bostanci, E., & Mishra, A. (2022). Sentimental Analysis of Twitter Users from Turkish Content with Natural Language Processing. *Computational Intelligence and Neuroscience*, *2022*.

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference on Web and Social Media*, *11*(1), 512–515.

Eggheads have found a positive link between the number of racist tweets and the number of racist hate crimes in US cities • The Register. (n.d.). Retrieved December 2, 2022, from https://www.theregister.com/2019/06/26/twitter_racism/

Gitari, N. D., Zuping, Z., Damien, H., & Long, J. (2015). A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, *10*(4), 215–230.

Hodeghatta, U. R. (2013). Sentiment analysis of Hollywood movies on Twitter. *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)*, 1401–1404.

Lin, H., Ma, J., Cheng, M., Yang, Z., Chen, L., & Chen, G. (2021). Rumor detection on twitter with claim-guided hierarchical graph attention networks. *ArXiv Preprint ArXiv:2110.04522*.

Ramya, M., & Pinakas, J. A. (2014). Different type of feature selection for text classification. *International Journal of Computer Trends and Technology*, *10*(2), 102–107.

Twitter Sentiment Analysis | Kaggle. (n.d.). Retrieved December 2, 2022, from https://www.kaggle.com/datasets/arkhoshghalb/twitter-sentiment-analysis-hatred-speech?select=train.csv

Venkatesh, B., & Vishwas, H. N. (2021). Real Time Sarcasm Detection on Twitter using Ensemble Methods. *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, 1292–1297.

Warner, W., & Hirschberg, J. (2012). Detecting hate speech on the world wide web. *Proceedings of the Second Workshop on Language in Social Media*, 19–26.

Wen, T., & Zhang, Z. (2017). Effective and extensible feature extraction method using genetic algorithm-based frequency-domain feature search for epileptic EEG multiclassification. *Medicine*, *96*(19).