Analysis of Tweets on 2020 Presidential Elections and Hate Speech Detection.

Rushikesh Gawande
Sumeet Sarode
Jashjeet Madan
Indiana University, Bloomington

May 3, 2020

Submission for ILS Z639 Social Media Mining, Spring 2020

Abstract

Twitter is a very important aspect of social media and is used vastly in day to day life by millions of people. Our study used twitter as a platform to study the ideas, expressions, and opinions of people regarding the upcoming 2020 Presidential election. This study aims to analyze the sentiments of the tweets related to these elections and build a machine learning model to classify the speech as hate speech or not. Further, topic modeling is performed on this corpus to analyze the tweets. We will discuss in great detail the prior assumptions in the study, the methodology used for data collection and data preprocessing, and performing sentiment analysis on tweets, the algorithms implemented to detect hate speech along with the implementation of topic modeling on this data.

Keywords: **Twitter, 2020 Presidential elections, hate speech, natural language processing, sentiment analysis, topic modeling**

Introduction

Twitter is a widely used social media platform and a microblogging service. It is used by people, different firms, organizations, news channels as well as government institutions to express their ideas, opinions and to access information, and share it with the world. According to Wikipedia, it has been found that the approximate number of active users on Twitter since 2018 is more than 321 million. Starting from 2006, today, Twitter has become an inseparable part of most people's lives and has expanded to be the most critical component of social networking in the world. It is used widely for social research as well as to study the behavior of people along with how they respond or react to specific intricacies and events. Specifically, because of this reason, along with the availability of Twitter API to gather required tweets, the decision was made to use Twitter data to conduct this research.

Political conversations and debates have always been a hotspot for people. Individuals from across the globe express their views and opinions and feel instrumental in identifying the issues. One of the significant political events is the Presidential elections of the United States of America. It is an undeniable fact that Twitter played a huge role in the 2016 elections. It was observed that around 1.4 million people from all across the world expressed their viewpoints through tweets with election-related labels in them.

With the 2016 Presidential elections, the focus of many things shifted. There was an eruption in sectarian, bigoted, racist, and offensive remarks. With anonymity on the internet, many of these were related to a particular sentiment because of provocative and incendiary discourse. 'The Anti-Defamation League documented the rise in anti-Semitic attitude, and tweets targeting journalists and politicians along with the people on either side of the line. From August 2015 to July 2016, the ADL found 2.6 million tweets with anti-Semitic language, which was during the campaign phase.' [12] Hence, it becomes imperative to follow the 2020 campaign period six months before the final vote is cast. Twitter provides an ample opportunity for this with a virtual goldmine of data synonymous with the public inclination.

Literature Survey

With the presidential elections in November 2020, the outpouring of hot debates, disagreements, controversies, and hate speeches gain massive rise. Typical hate speeches during the election time differ from the recurrent tweets that go around the internet. With frequent disagreements among various socio-political groups as the time nears, social media platforms provide an excellent opportunity to study these trends. Their ability to keep an account of the tweets makes them an invaluable source to study crowd reasoning. Crowdsourced data processing is a participatory process of appropriating a large number of people to create a dataset. They are the primary suppliers to supplement the service with specific details, incomplete or existing information, and annotation of data.

'Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior by Antigoni-Maria Founta et al.' is one such example, that provides a comprehensive behavior-study of Twitter for eight months. 'They investigated a wide variety of labeling schemes, which cover different forms of abusive behavior. They leveraged the power of crowdsourcing to annotate a large set of abuse-related labels.' [8] The dataset was made public for further study, and we have utilized similar datasets for training our models. An empirical analysis of similarities produced a set which is a general representation of all the types of hate speech. This set was annotated to obtain the final structure. Some less significant labels, which included Cyberbullying, were discarded.' [8]

Many such datasets have been curated in the past, along with various experiments. These ranged from standard lexicon-based root word extraction to determine the sentiment of the tweet, to statistical models that study different parameters and correlation among several parameters. One such paper that specifically studied the labels that were dropped in the previous paper is Mean Birds: Detecting Aggression and Bullying on Twitter by Despoina Chatzakou et al., which focused on cyberbullying and aggression. 'The basis of their methodology was on extracting text, user, and network-based attributes, studying the properties of cyberbullies and aggressors, and the salient features that distinguish them from regular users. Their structure included post frequency, participation in online communities, and popularity.' [7]

"Mr. Trump claims he is surprised his election has unleashed a barrage of hate across the country," said Michael Cohen, Trump's lawyer. 'The Effect of President Trump's Election on Hate Crimes by Griffin Edwards et al. used time series analysis and panel regression techniques to examine the relationship between President Trump's election and trends in reported hate crime rates at the national and local level from 1992 through 2017. They hypothesize that it was not just Trump's racial language that prompted hate crimes to intensify during the election cycle. Instead, they concluded that it was Trump's eventual election as President of the United States that may have justified this discourse in the eyes of perpetrators and intensified the increase in hate crime. Building over that, 'Racial Bias in Hate Speech and Abusive Language Detection Datasets by Thomas Davidson et al. measures racial bias in hate speech and abusive language detection datasets. This study was compelling as the society discriminates against the groups they studied who are often the targets of the abuse.' [9]

Bias in machine learning is described as the evaluation of outcomes that are systematically prejudiced because of the wrong hypotheses. Bias may cause the models to distinguish aggressively against the same groups that they are intended to protect. The authors presented an exhaustive study that consisted of five datasets.' The first dataset had 130k tweets that contained at least one of the seventeen hateful phrases. The second dataset consisted of 16,849 tweets labeled as either racism, sexism, or neither. To account for any bias, 2876 tweets in the dataset were relabeled, along with new samples from the tweets collected initially, to serve as the third dataset. The fourth category had tweets that exhibited racism *and* sexism. The fifth dataset had tweets containing terms from the hate speech database, which is crowdsourced. This dataset had 24,783 tweets annotated as hate speech, offensive language, or neither.'[9] Overall, this is the most crucial dataset for training our model.

The ultimate purpose of developing machine learning models is to automate some processes. In this case, it is automatic hate-speech detection on social media for the separation of hate speech from other instances of offensive language. A similar paper by Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber in Automated Hate Speech Detection and the Problem of Offensive Language presented with different lexicon-based methods, their limitations, and an improved machine learning model to distinguish hate speech from other tweets. As previously shown, 'Bag-of-words approaches tend to have high recall but lead to high rates of false positives since the presence of offensive words can lead to the misclassification of tweets as hate speech' (Kwok and Wang 2013; Burnap and Williams 2015). 'The authors also included binary and quantity indicators for hashtags, mentions, retweets, and URLs, as well as features for the number of characters, words, and syllables.' [11]

Data Gathering

The most significant aspect of research of such a magnitude is the gathering of data. Various social media platforms provide an opportunity for data mining and extraction. There are various approaches to this, such as web crawlers, web scraping, automatic bots, etc. One needs to keep in mind ethical considerations and abide by laws to make sure they do not violate any privacy laws or policies. Tweepy is an easy-to-use Python library for accessing the Twitter API. [12] This API (Application programming interface) is used to facilitate the extraction of a small sample of tweets from the total tweets for academic or research purposes. This official tool, which is provided by Twitter, also offers the developers access to numerous features such as upvotes, downvotes, geolocation, likes, direct messages, amongst other features. Tweepy handles the authentication and the connection setup required to stream the Twitter data and enables the user to gather tweets from Twitter only after the proper authentication is completed.

To collect those tweets that were posted for the 2020 Presidential election, we first obtained various hashtags that were used for posting about the election. The following hashtags were used on twitter for the 2020 election: i. #2020election ii. #2020elections iii. #2020_presidential_election iv. #Election2020 v. #presidentialelection. After determining the hashtags, we collected all the tweets that contained any of the above mentioned five hashtags by employing the search API integrated with the Tweepy library of python.

Our goal was to determine whether a given tweet falls into the category of a regular tweet or a tweet possessing hatred. We built various machine learning models, trained them, and tested them on the tweets collected by us. To train these models, vast training data is required, which consists of a considerable number of data samples. To get the training data, we gathered various hate speech datasets that are publicly available on the Kaggle website. These are, i)Hate speech detection|NLP ii)Hate_speech_dataset iii)Twitter hate speech, iv)Hate-Offensive Speech Detection v)Hate Speech. All these datasets are available in different formats with various attributes and possibly superfluous information. To transform the data into the required format, all the files were merged to form a single format. In the end, we had a combined dataset with over 130,000 annotated tweets that were labeled as hate speech or not. This dataset was used to train the model.

The collected tweets, which were to be classified as hate speech or not, were first cleaned and preprocessed. In cleaning, we removed all the punctuation, URLs, hashtags, and stop words. Stop words in any language can be defined as the most frequently used words, which are often irrelevant for analysis.
The steps include:

1. Removing any HTML content such as URLs.
2. Removing punctuation by using the built-in string library in python.
3. Tokenizing to break the strings into a list of words (n-grams) by using RegEx (Regular Expression).
4. Removing stop words by using the NLTK package's built-in function: 'stopwords'.
5. Lemmatization or Stemming to shorten words back to their root to derive its meaning by eliminating prefixes or suffixes.

Methodology

Data Preprocessing: The data that was gathered to train the machine learning model was preprocessed before loading into the model. Stemming and Lemmatization are popular methods to extract the meaning of the words by omitting tenses, prefixes, and suffixes. These new words, whether inflected or root form, are considered as a single entity. In our case, this was achieved by performing Lemmatization on the tweets. In Lemmatization, all the words are converted to their canonical (original) form called "lemma" with the help of the Natural Language Toolkit (NLTK) module in python. It is the most efficient python library for processing textual data to extract various features from a given sentence in natural language.

Sentiment Analysis: Sentiment Analysis is also known as Opinion Mining. It is the process of finding the degree of positivity or negativity of a given sentence or corpus. After data preprocessing, the sentiment of each tweet was determined with the help of the polarity score of the tweet calculated using the Sentiment Intensity Analyzer module in NLTK. This is based on the lexicon-based VADER module. Vader stands for Valence Aware Dictionary and sEntiment Reasoner which labels each word or a group of words as to how positive, negative, or neutral they are and ultimately calculates the polarity of the sentence or corpus as a whole.[6] VADER first splits the entire corpus into tokens. A token, in this case, can be defined as words. It then calculates the polarity of all the tokens in the sentence to find the overall positivity and the negativity of the sentence. The result of sentiment analysis is a value between -1 to 1, with -1 being the most negative and +1 being the most positive sentiment.

Overview of the ML models: In this study, the association of tweets with hate speech or not is binary classification, which belongs to the category of supervised learning. So, the labeled data of 130,000 tweets we collected from various datasets in its entirety was used for training and validating our algorithms. Subsequently, these models were used to detect hatefulness in the scraped tweets of the 2020 presidential elections. With several algorithms in the world of machine learning to perform the task of classification, there is no straightforward interpretation as to which one is the best amongst them. It all depends on the nature of the data set, the size of the dataset, features present in the data, and the hyperparameters of the algorithm. The algorithms which we used are eager learners i.e., the judge based on the data they have been trained on. They receive the actual data to classify after the training and the validation stage. Hence, eager learners tend to take a long time to train; nonetheless, they need lesser time to predict. The four algorithms we used are explained below.

1. **Logistic Regression Classifier:** The logistic regression algorithm is a model that predicts the target class to which the data belongs based on the hypothesis. It uses the sigmoid function to estimate the probability of how similar the predicted value to the actual value is. There is a threshold set and this estimated probability is classified depending on this threshold. It is quite a simple and efficient algorithm with low variance and the cost function used is a convex function so that the gradient can converge into the global minimum. We used it as a baseline model, and it was implemented using the models package in the sklearn library. The solver was specified to be lbfgs and the penalty was set to use ridge regression.
2. **Decision Tree Classifier:** The decision tree is a supervised machine learning algorithm that can be used for both classification and regression purposes. They are most commonly termed as CART (Classification and Regression Trees). The data is divided into small subsets to generate a tree structure. Each node represents a criterion for the split and its branches are the possibilities of the outcome for that criteria. It can handle both categorical and numeric data and the non-linear relationships amongst features do not affect the tree. The depth of the tree is directly proportional to the complexity of the model and large depth may lead to overfitting. The algorithm was implemented using the tree package in the sklearn library. The split criterion was set to Gini-index and the max depth for the key was set as default, which is none, i.e. the tree is expanded until all the leaf nodes are class labels.
3. **Gradient Boosting Classifier:** The boosting technique is used to enhance the efficiency of decision trees. Instead of just one tree, multiple decision trees are constructed and then merged to achieve optimum performance. The ensemble is built by training the multiple models such that each model is trained to rectify the errors of the previous one. So, it builds a strong and efficient model by combining weak and simpler models. The loss function used was deviance and the cost function was set to friedman_mse.However, in our case, the gradient boosting did not outperform the decision trees and gave results a little less accurate than decision trees.
4. **Neural Network Classifier:** The neural network is a layered network consisting of neurons and is used for both classification and regression. The architecture of the neural network consists of an input layer, hidden/dense layers, and an output layer. The number of neurons in the input layer is equal to the number of features in the data and the number of neurons in the output layer depends on the number of classes in the target variable. The number of hidden layers and the number of neurons in each hidden layer is variable. There are weights and biases associated with neurons and each hidden layer along with the output layer has an activation function. We build the neural network with one input layer, one hidden layer, and one output layer. The hidden layer has 100 neurons with activation

function as relu. The output layer has 2 neurons with softmax as the activation function. The optimizer was set to Adam and the learning rate was 0.001.

Topic modeling: The topic modeling is one of the techniques of unsupervised learning, used to cluster the word groups that are similar to each other. It is implemented mainly to analyze the huge data from social media platforms, emails, open-ended survey responses, etc. The main aim of topic modeling is to find topics out of the documents based on the similarity between the occurrences of words. It monitors the correlation between the frequency of words occurring in the topics and patterns in those words. LDA is a prominent algorithm used to implement topic modeling. After doing all the preprocessing on the tweets, the LDA package of the sklearn module was used to implement the algorithm. The whole tweets were clustered to form topics that are a collection of most relevant keywords which will tell everything that the topic is all about.

Results and Discussion

Performing Sentiment analysis on the cleaned tweets revealed that 41% of the tweets collected had a positive sentiment i.e. with a polarity score greater than 0. The tweets with a polarity score of less than 0 summed up to 29% of the total tweets. It was observed that the polarity scores of both- tweets with a positive sentiment and those with negative sentiment were spread throughout the extremities of +1 and -1. The mean score of positive tweets was 0.47 while that of negative sentiment tweets was -0.43

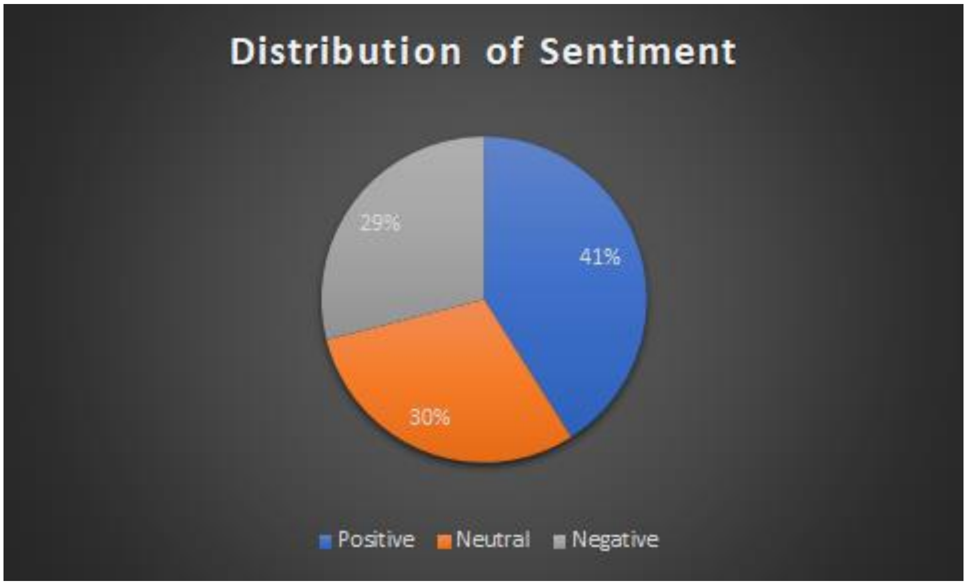

Fig 1.1: Distribution of sentiments**.**

To determine if the gathered tweets were normal or hate speech, we implemented four different machine learning models namely- Logistics Regression, decision tree classifier, Gradient Boosting Classifier, and Multi-Level Perceptron (Neural Network) Classifier. Amongst these models, Neural Network was chosen for making the predictions based on the accuracy metric- F1 score and accuracy score. Since the initial analysis of the tweets gathered for training showed that it was an imbalanced class data, using accuracy score alone was not a sufficient metric for determining the efficiency of the model. Since the F1 score takes into consideration both- precision, and recall of the model, which are very essential when the data is not evenly distributed among the target classes, it was considered along with the accuracy score to measure the efficiency of the models. The accuracy score is calculated as the percentage of the data samples the model classified correctly. The precision of a model can be defined as the proportion of the data samples our model predicted positive were positive. Recall on the other hand is the ability of the model to determine the positive samples. F1 score which considers both precision and recall can be calculated by the following formula:

$$F_1 = 2 * \frac{precision * recall}{precision + recall}$$

Logistic Regression was implemented as the baseline model. The accuracy of Logistic Regression was 89.58%. Out of 100 tweets classified by the model as being normal speech, 84 tweets were normal. In the case of hate speech out 100 tweets recognized by the model to be hate speech, 94 were correct. For every 100 tweets given to the model belonging to each of the normal class and hate speech class, the model correctly classified 88 and 91 tweets for the respective classes. The F1 score for the normal tweets came to be 0.88 and that for hate speech tweets was 0.91. It classified 89.58% of the total tweets correctly. These results obtained were set as our baseline results.

In an attempt to improvise the results obtained by implementing Logistic Regression, Decision Tree Classifier was implemented. This model had an edge over Logistic Regression when it came to classifying hate speech tweets correctly. But it misclassified 13 normal tweets into the hate speech category. It was found that out of 100 tweets detected as hate speech, only 91 were correct. While only 86% of the tweets classified as normal were correct. Normal tweets were classified with an F1 score of 86% while that for hate speech was equal to that of Logistic Regression that is 91%. The overall accuracy of the Decision Tree Classifier was close to that of Logistic Regression which turned out to be 89%.

To improve the classification, Gradient Boosting Classifier was implemented. Since it groups various weak models such as decision trees, the performance of the Gradient Boosting Classifier is expected to be better than simple models in theory. But implementing the model on the tweets dataset gave a low performance on the test data. This was caused mainly due to overfitting, a scenario in which the model trains itself rigorously on the training data to give a poor performance on unfamiliar (test) data. The F1 score for normal tweets for this classifier was lower than the previous two models which were 0.84. Hate speech tweets were also classified with a comparatively low F1 score of 0.86. Amongst the 4 models implemented, Gradient Boosting Classifier produced the least number of correct predictions which were around 85.3%.

The model that classified the maximum number of tweets correctly in the test data was Neural Network (Multi-Layer Perceptron Classifier). The accuracy of this model on test data was 90.42% which was the highest among all the models. For 100 tweets classified as hate speech, 92 tweets were correct classifications. This number was around the same in the case of normal tweets. The model made an average of 90 correct classifications for every 100 normal class predictions. It classified out 100 tweets recognized by the model to be hate speech, 94 were correct. This model had a high recall rate too. It correctly recognized 87% and 93% of normal and hate speech respectively from a given group of data samples of each class. The F1 score for normal tweets and hate speech was 0.88 and 0.92, respectively.

Algorithm	Accuracy %	Precision		Recall		F1 Score	
		Normal	Hate Speech	Normal	Hate Speech	Normal	Hate Speech
Logistic Regression	89.58%	0.84	0.94	0.92	0.88	0.88	0.91
Decision Tree	88.97%	0.86	0.91	0.87	0.90	0.86	0.91
Gradient Boosting	85.26%	0.74	0.98	0.97	0.77	0.84	0.86
Neural Network	**90.42%**	**0.90**	**0.92**	**0.87**	**0.93**	**0.88**	**0.92**

For getting a more clear idea about the performances of all the models, Area Under the Curve was plotted for all the models. It is the measure of how the recall value of the model (true positive rate) and the false positive rate of the model operate with each other. It is the combined measure of all the accuracy metrics for a given model and the dataset.

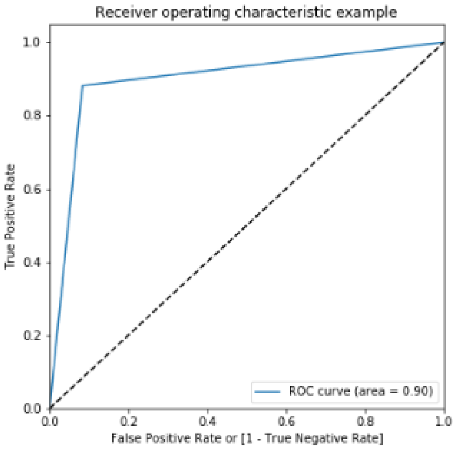

Fig 1.2 AUC for Logistic Regression

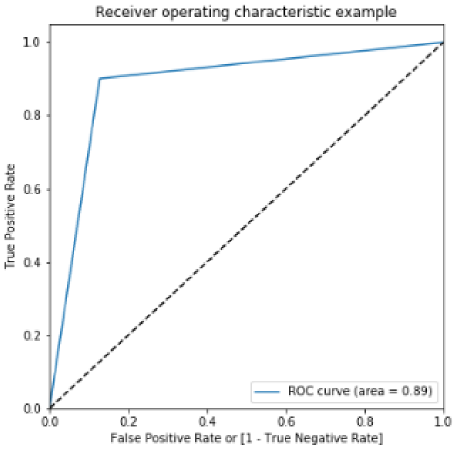

Fig 1.3 AUC for Decision Tree

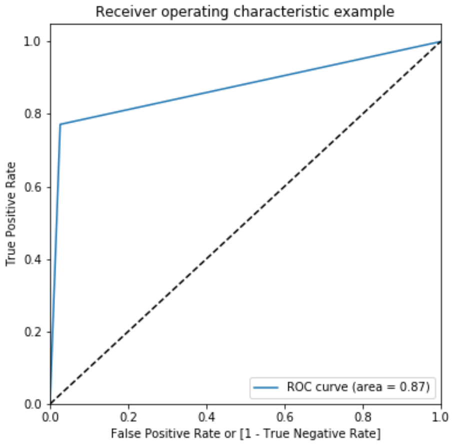

Fig 1.4 AUC for Gradient Boosting

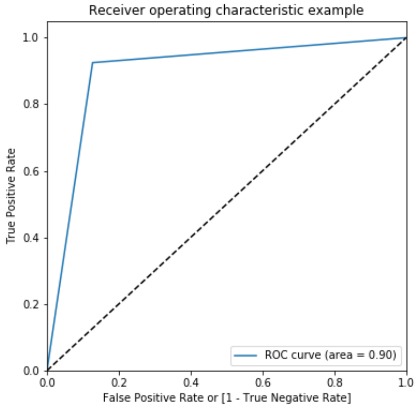

Fig 1.5 AUC for Neural Network

After analyzing the metrics of all the models, Neural Network stood out amongst all the other models. This model then was used for the task of detecting the hate speech in the tweets collected for the Presidential election of 2020. Out of the 29,000 tweets collected, the model recognized 12,000 tweets as hate speech, and 17,000 were classified as regular tweets.

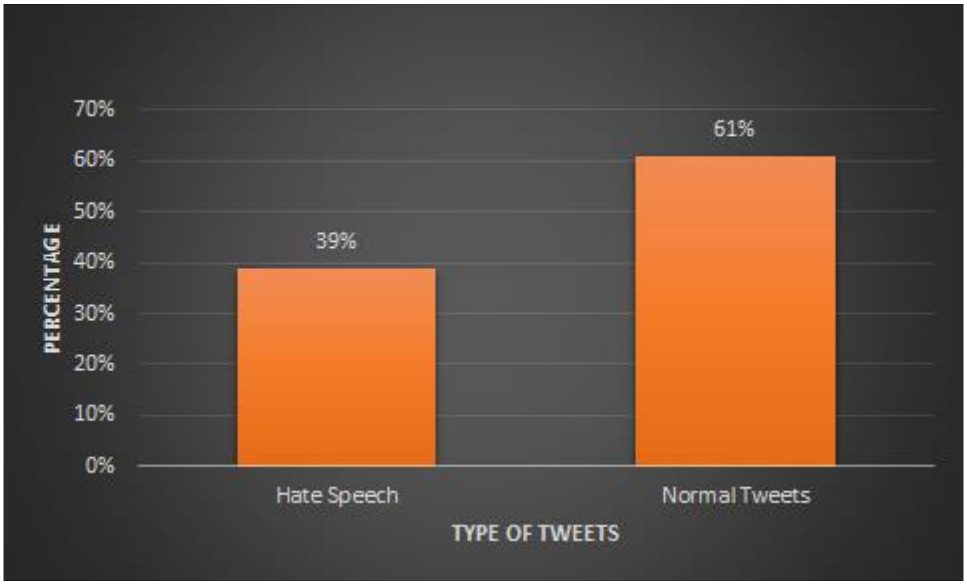

Fig 1.6 Distribution of hate speech

From fig 1.6, it is clear that hate speech constituted approximately 39% of the tweets posted for the discussion of the 2020 election. The remaining 61% of tweets collected were regular posts.

The topic modeling was performed on this corpus and it gave us the following results by finding 7 topics from the given corpus. Now the words were grouped into these seven topics based on the relevance along with similarity i.e. how closely these words resemble each other. The seven topics along with the top 15 words from each topic have been shown in the below table.

	Top 15 Words
Topic #1	trump, failure, httpst, exactly, httpstcojxiwfghdcl, curveball, ad, new, af, afbranco, branco, cartoon, election, 2020, rt
Topic #2	findings, approve, monday, home, ohio, key, right, vote, poll, joebiden, needs, ritamollerpalma, state, realdonaldtrump, rt
Topic #3	biden, need, support, time, republicans, political, president, china, new, Americans, coronavirus, amp, trump, realdonaldtrump, rt
Topic #4	voters, GOP, election, win, president, joe, biden, pollsofpolitics, retweet, november, voting, joebiden, realdonaldtrump, vote, rt
Topic #5	ppl, said, mailin, sumoh7, demswork4usa, says, support, epochtimes, core, idea, presidential, don, think, candidates, rt
Topic #6	cash, month, people, tune, hand, states, lot, house, look, joe, president, biden, just, trump, rt
Topic #7	cognitivediss00, tara, appolitics, primary, wins, breaking, trump, democratic, people, presidential, vote, democrats, joe, biden, rt

Table 1.1

Conclusion

A close investigation of the tweets illustrates that we can successfully separate offensive content from other content. Although the internet is filled up with negative content, Presidential elections are, in particular, infamous for spreading hate. The results are consistent with the previous studies conducted around the 2016 Presidential Elections. Sentiment analysis with the help of natural language processing on the collected dataset presented crucial insights on how peoples' opinions vary while considering the elections. The implementation of machine learning models can be leveraged successfully to recognize hate speech on Twitter in real-time, which is a key challenge for Twitter. Furthermore, for digging out the hidden semantic structures of the tweets a statistical model was implemented using LDA to find the topics occurring in the dataset along with the words associated with those topics.

We hope to see in future research and studies the various applications of automating hate-speech detection. Analysis of the social dynamics in addition to the events and the interactions between the group of people in which hate speech happens more closely. Also, hate speech has many forms, and it could prove valuable to distinguish amongst them as racism, sexism, slang offenses, and vile defamations. Also, deep learning is a hot topic that can be utilized for understanding the underlying semantics with the help of Recurrent Neural Networks and Long Short-Term Memory models.

References

[1] Wikipedia - Twitter https://en.wikipedia.org/wiki/Twitter.
[2] Tweepy documentation: http://docs.tweepy.org/en/latest/getting_started.html#api.
[3] NLTK toolkit documentation: https://www.nltk.org/index.html.
[4] Cihon P and Yasseri T. *A Biased Review of Biases in Twitter Studies on Political Collective Action. Front. Phys. 4:34.* 2016.
[5] https://scikit-learn.org/stable/supervised_learning.html#supervised-learning
[6] https://www.geeksforgeeks.org/python-sentiment-analysis-using-vader/
[7] Chatzakou D, Kourtellis N, Blackburn J, Cristofaro E, Stringhini G, Vakali A. *Mean Birds: Detecting Aggression and Bullying on Twitter*. WebSci '17: Proceedings of the 2017 ACM on Web Science Conference (13–22). 2017.
[8] Founta A, Djouvas C, Chatzakou D, Leontiadis I, Blackburn J, Stringhini G, Vakali A, Sirivianos M. *Large Scale Crowdsourcing and Characterization of Twitter Abusive Behavior*. Twelfth International AAAI Conference on Web and Social Media. 2018.
[9] Davidson T, Bhattacharya D, Weber I. *Racial Bias in Hate Speech and Abusive Language Detection Datasets.* Proceedings of the Third Workshop on Abusive Language Online (25–35). 2019.
[10] Edwards G, Rushin S. *The Effect of President Trump's Election on Hate Crimes.*
[11] Davidson T, Warmsley D, Macy M, Weber I. *Automated Hate Speech Detection and the Problem of Offensive Language.* ICWSM. 2017.
[12]https://www.usatoday.com/story/tech/news/2016/10/21/massive-rise-in-hate-speech-twitter-during-presidential-election-donald-trump/92486210/