**Twitter Text Sentiment over Mask Mandate**

**Tanzil Baraskar1, Hafsa Nadim2,and Tenzin Tashi3**

1Tanzil Baraskar: TBarask000@citymail.cuny.edu

2Hafsa Nadim: HNadim000@citymail.cuny.edu

3Tenzin Tashi: TTashi001@citymail.cuny.edu

**Abstract:** Twitter is becoming one of the major online platforms for expressing opinions and thoughts. As people share more and more opinions online, we believe that the sentiment analysis of online communication has become a new way to gauge public opinions of events and actions in the world. In this project, we will be using Natural Language Processing to perform sentiment analysis by classifying the degree of emotional valence in the text from Twitter’s tweet data.

# 1 Introduction

Sentiment analysis has also been referred to as opinion mining. It is a natural language processing technique that is used to determine whether the data is positive, negative, or neutral. Nowadays, the Internet has changed the way people express their views and opinions. People use many different social media platforms such as Twitter, Facebook, and Instagram to express their thoughts and opinions on topics they are passionate about which could lead to debates and discussions, resulting in social media to become an indispensable platform in people’s daily lives. These platforms are generating a wide range of sentiment data through tweets, posts, blogs, comments, and reviews.

Twitter is one of the largest social network platforms available and it hosts user-generated posts. It serves as a large basis for the world to get updates on local and international news around mask mandates. In this project, we will implement a Twitter sentiment analysis model that can help overcome the issue of identifying the sentiment of the tweets.

For this classification experiment, two types of sentiments will be utilized around positive and negative sentiments. In order to obtain higher accuracy, there are two prevailing techniques for large-scale analysis: machine learning and lexicon-based approach. The machine learning approach can automatically detect the polarity of texts, however it requires supervised training on labeled data sets. The data set we will be using will be put into a NLTK model. NLTK is a Python library that provides a set of diverse natural language algorithms. It consists of the most common algorithms such as tokenizing, part-of-speech tagging, topic segmentation and sentiment analysis. The lexicon-based approach uses sentiment dictionaries with opinion words and uses this to match them with the data to determine the polarity.

# 2 Related Work

In [1], a survey is conducted on techniques for opinion mining and it uses algorithms and models such as Support Vector Machine, Naive Bayes and Max Entropy to deal with Twitter data streams. The survey used a Twitter dataset from Stanford, where analyses were done using different feature extraction techniques. The different machine learning models trained the dataset with feature vectors. When discussing the machine learning approach, it can be subdivided into two of the following categories which are supervised and unsupervised learning. Supervised learning is based on a labeled dataset and these are trained to get meaningful outputs. Machine learning plays a critical role in the accuracy and success of sentiment classification of Twitter data. Machine learning models and techniques such as Support Vector Machines, Maximum Entropy and Naive Bayes are used to classify Twitter tweets into classes and have achieved success in sentiment analysis. In the survey, the results show that the SVM outperforms Naive Bayes and Maximum Entropy. The best result was obtained from SVM when it was used on a feature set of a combination of Unigram and Bigram with stopword removal. In Umer, et al (2020) [2], the proposal centers around performing sentiment analysis on Twitter airlines sentiment, women’s e-commerce clothing reviews, and on hatred speech detection. The approach is done with the use of a neural network and a long short-term memory network. Two feature extraction methods known as term frequency-inverse document frequency and word2vec are used to evaluate their efficacy and the performance of the approach is analyzed against a large set of machine learning classifiers such as the Support Vector Classifier, Stochastic Gradient Descent, Random Forest, Voting classifier, and Logistic Regression. The results show that the Voting classifier and the Stochastic Gradient Descent achieve high accuracy results when term frequency-inverse document frequency features are used for classification for all three datasets. Stochastic Gradient Descent performs better than three of the classifiers which are the Voting classifier, Random Forest and the Support Vector Classifier, when it comes to precision and recall. We can see that classifiers, Stochastic Gradient Descent and the Voting classifier are a good fit for achieving high accuracies. This proposal about women’s e-commerce clothing reviews is similar to our mask mandate project because both for both, the reviews/attitude help companies and businesses make decisions.

In [5], the proposal approach uses supervised statistical machine learning methods to build News and Social Sentiment from Bloomsberg. Here, the human expert manually assigns a positive, negative or neutral score to each news story or tweet and then the data is fed into a machine learning model, the Support Vector Machine. The model automatically assigns a probability of being positive, negative or neutral to each news story. The result is that the sentiment strategies outperform the corresponding benchmark index ETFs. In [6], the research proposal’s objective is to observe how the world views Dubai and this is done by classifying news tweets based around the sentiment of positive, negative and neutral. The Naive Bayes Algorithm and Long Short-Term Network are used for this proposal project. To label the sentiment, three machine learning techniques, TextBlob, VADER and SentiWordNet are used to perform sentiment analysis. TextBlob is a Python library for processing textual data, VADER provides for the intensity of the emotion and determines whether a tweet is positive or negative and SentiWordNet uses the WordNet database. WordNet is a database for words included in the English language and it is organized by meaning and concept. The results are that most of the news tweets centered around Dubai are positive and neutral. The Long Short-Term Network obtained a better accuracy of 89%, while Naive Bayes multinomial had an accuracy of 71%. From this research, we understand that techniques such as TextBlob, VADER, and SentiWordNet can be used to aid in the process of determining the valency of a tweet. After a supervised classification technique is chosen, it is important to select the correct features. The most commonly used features in the sentiment classification are: part of speech information, opinion words and phrases, term presence and their frequency, and negations.

The lexicon-based method uses sentiment dictionaries with opinion words and it relies on a sentiment lexicon. The work in [3] proposes a survey of existing sentiment analysis approaches in fields such as health, riots, air pollution, stock sales, and disaster management. The approaches are to be categorized into the following: machine learning based, lexicon-based, hybrid-based (machine learning and lexicon-based), or graph-based. With the machine learning approach, the SVM LibLinear model is used to detect the emotions on Twitter messages. The result of this is that the SVM’s classifier accuracy is 98%. The lexicon-based approach developed a framework to classify movie reviews into positive, negative and neutral words. The results show that the proposed lexicon-based method was able to classify sentiment with 52% accuracy. The hybrid approach deals with the preprocess and re-label tweets using a weight-based classification. The results were that the pre-processing and drift detection techniques significantly improved the classification accuracy by 70%. The graph-based approach deals with studying Twitter users disseminating information during the Australian floods to reveal interesting patterns. A graph was built using three types of relations between retweets, tweets published by the same person, and tweets replied to. The result was that the accuracy of the graph-based optimization approach was 68.3%. This research allows us to understand that there are different approaches that can be used to improve accuracy. Kouloumpis, et all (2011) [4], mentions a variety of features representing information from a sentiment lexicon and part of speech features. For their experiment, they used three different corporas of Twitter messages. They used the hashtagged data set, the emoticon data set, and a manually annotated data set produced by the iSieve Corporation. The hashtagged data set is a subset of the Edinburgh Twitter corpus. To create the hashtagged data set, the duplicate tweets need to be filtered and the distribution of hashtags need to be investigated. The experiment results were that using hashtags to collect training proved to be useful, as did using the data collected based on positive and negative emoticons. From this experiment, we learn how to filter tweets which can come in useful for the mask mandate tweets and like this experiment, our data also uses hashtags, which we use to extract live tweets.

# **3 Data**

**3.1 Collecting data**

In this project, we used two main datasets:-

* The dataset available at [kaggle](https://www.kaggle.com/competitions/tweet-sentiment-extraction/overview) which is composed of about 20k tweets to train sentiment predictors.

This dataset is used to train our model which finds the sentiment when a new tweet is given.

* The Twitter API. We extract the live tweets using specific hashtags and words that must be contained within the tweets we pull. For this particular project, we used the following keywords/hashtags: ‘covid19’, ‘quarantine’, ‘#nomaskmandate’, ‘#CovidIsOver’, ‘#novaccinemandate’

This REST API will extract and filter the specific tweets only on which we will run our sentiment analysis model.

**3.2 Preprocessing**

The following preprocessing was made on the kaggle training dataset:

* From all the tweets, first we remove the URLs

Some tweets contain links which do not help us at finding the sentiment of that tweet therefore we removed all the URLs.

* Tokenize text:

Tokenization is a process of breaking down a paragraph into a sentence, sentence into words, i.e., splitting the text into smaller units.

* Remove emails:

We found that some of the tweets in the dataset had emails which did not help at finding the sentiment of those tweets, same as the URLs, therefore we removed the emails from it as well.

* Remove new lines characters, single quotes:

Similarly, special characters like new lines, single quotes, etc are not an important feature in finding the sentiment therefore we decided to remove it.

* Remove all punctuation signs:

In this step, all the punctuations from the text are removed. The String library of Python contains some predefined list of punctuations such as ‘!”#$%&'()\*+,-./:;?@[\]^\_`{|}~’

* Lowercase all text:

In this step, all the text is converted into the same case preferably lower case. The reason for doing this step is because for example: ‘Hate’, ‘hate’ , ‘HATE’ all mean the same thing therefore we do not want our model to treat it as different.

**3.3 Building the label list**

As the dataset is categorical, we need to convert the sentiment labels from Neutral, Negative and Positive to a float type that our model can understand. To achieve this task, we used the *to\_categorical* method from keras.

* 0 for Neutral
* 1 for Negative
* 2 for Positive

**3.4 Data Exploration**

In the train dataset we have a total of 27,481 data out of which only 1 has a null value which was filled by the text “No content”. The dataset consists of 3 sections namely textID, text, and selected\_text. textID is the unique id given by the tweeter to each tweet, text is the actual tweet extracted from the tweeter and selected\_text is the text that we have got after doing the preprocessing.

**3.5 Graphs**

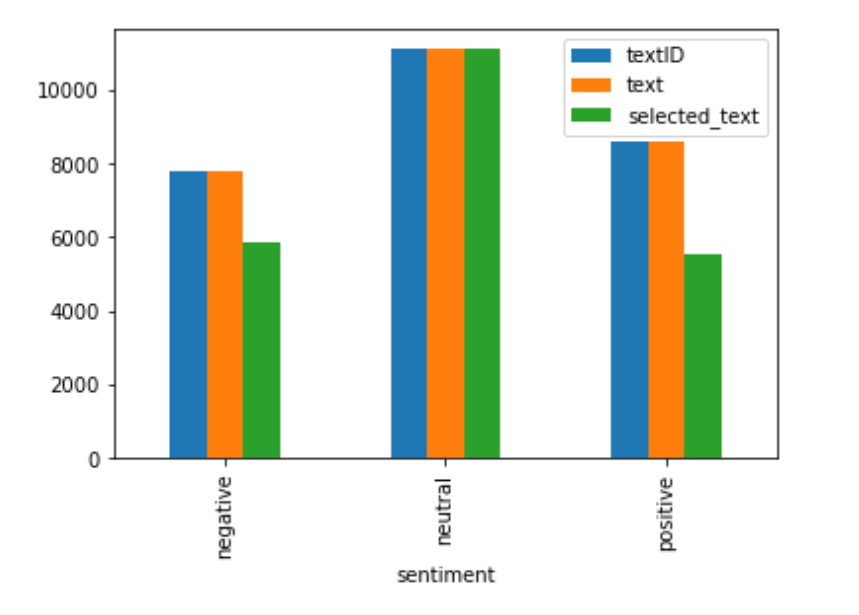
****

Fig 1: Here, we have the data distribution of positive, negative, and neutral.

In the above bar chart, we can see the data distribution of positive, negative, and neutral, are almost equal which means we are not biased toward any one of the sentiment.

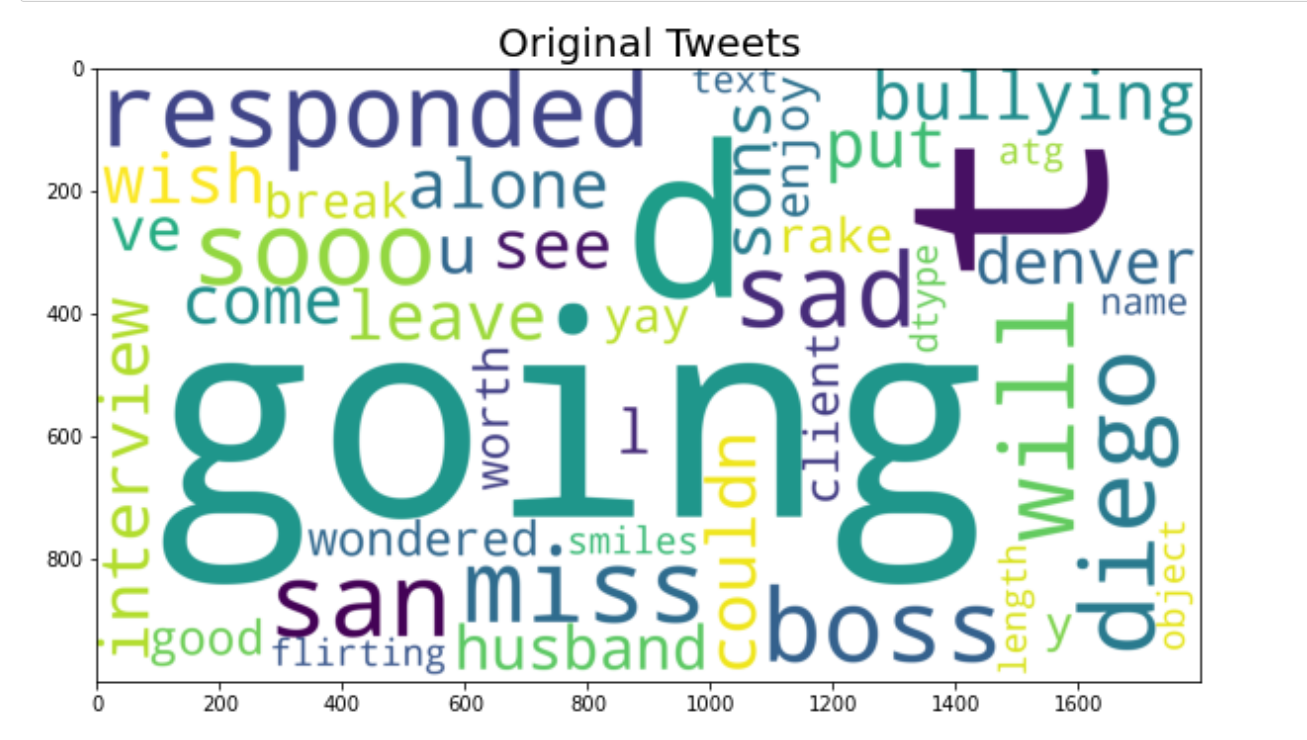


Fig 2: Word cloud formed by the original Twitter tweets

In the above word cloud, words like going, boss, miss, sad, etc frequently appeared in the tweets.

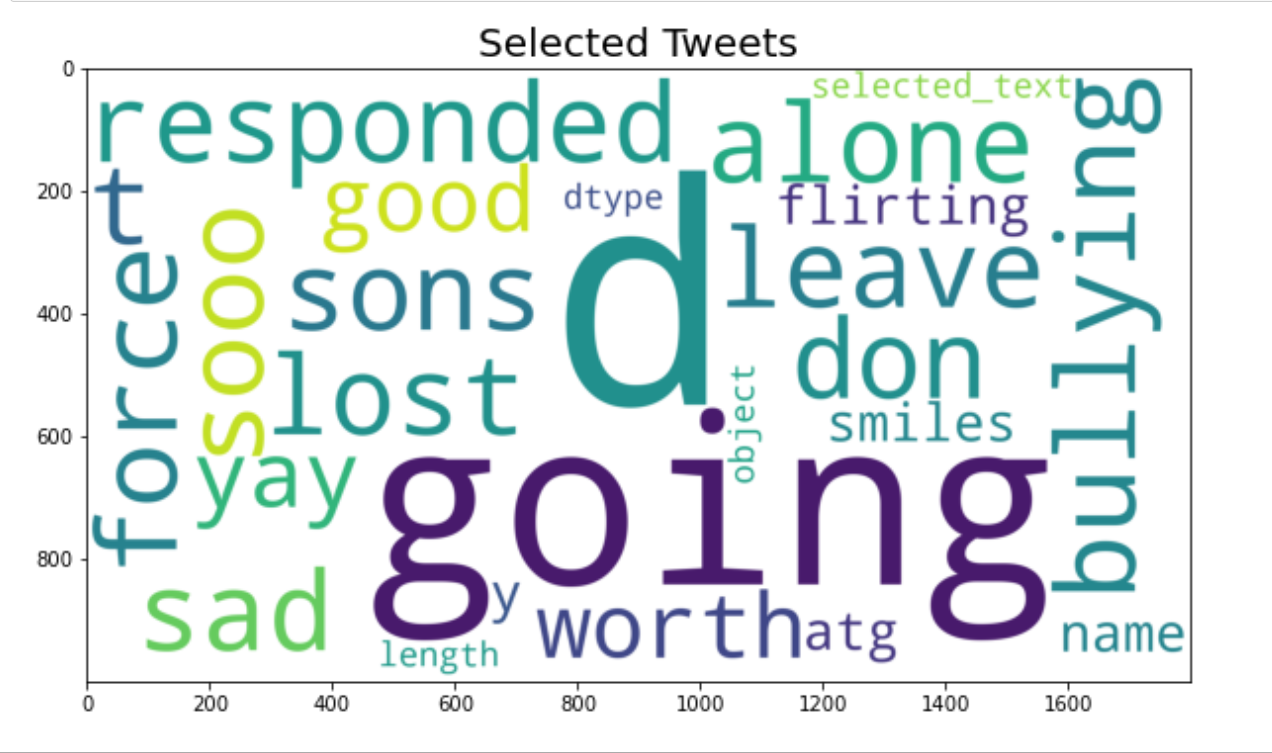


Fig 3: Word cloud of selected Tweets words

The word cloud of the selected tweets showing we have words like lost, leave, flirting, sons, good, sad, etc appeared a lot after preprocessing of the original tweets.

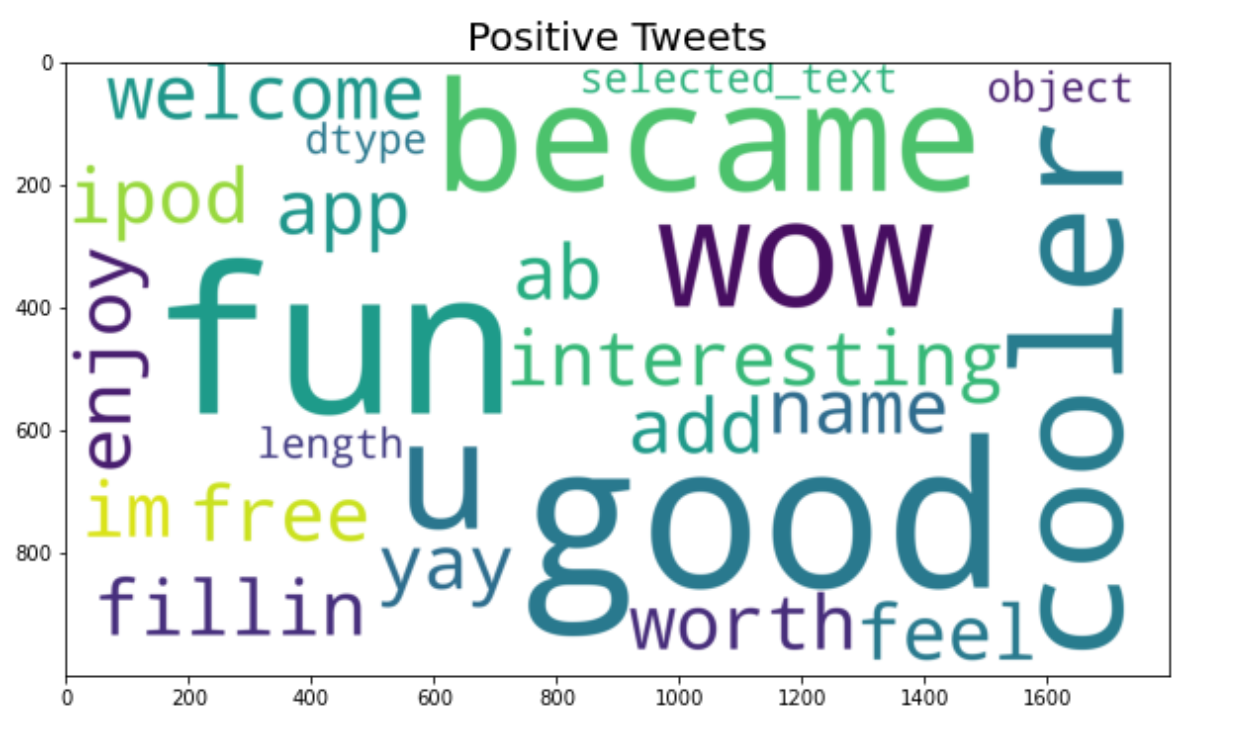


Fig 4: Word cloud formed by Positive words from the Tweets

The word cloud of the Positive tweets shows that words like fun, wow, good, interesting, cooler, enjoy, etc appeared a lot in the positive tweets.

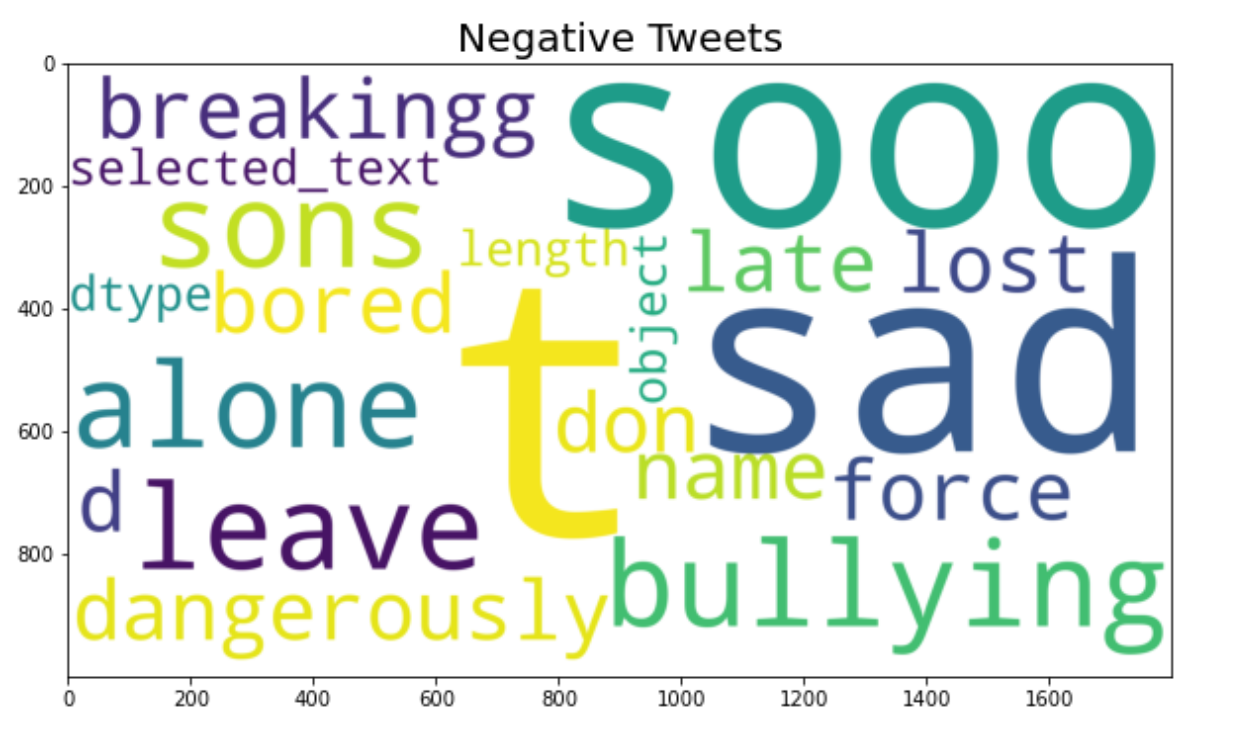


Fig 5: Word cloud formed by Negative words from the Tweets

The word cloud of the Negative tweets shows that words like sad, bullying, force, leave, alone, dangerously, etc appeared a lot in the negative tweets.

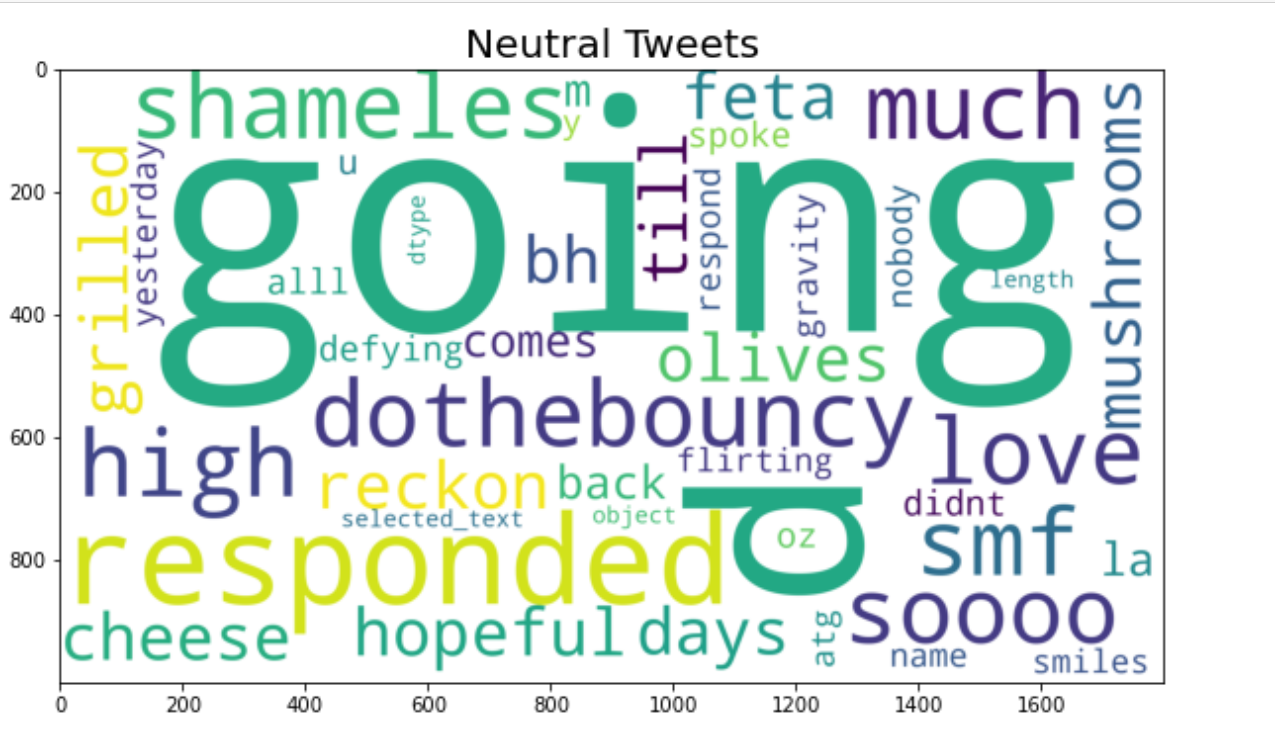


Fig 6: Word cloud formed by Neutral words from the tweets

The word cloud of the Neutral tweets shows that words like going, much, responded, days, cooler, cheese, etc appeared a lot in the neutral tweets.

# **4 Methodology**

# We implemented several algorithms on our current training dataset. The goal was to determine which Recurrent Neural Networks will perform the best and then apply it to our entire dataset. We implemented the following 3 Recurrent Neural Network Models and computed the accuracy for using the testing dataset:

* **LSTM (Long Short Term Memory) Model** 
  + **Bidirectional LSTM Model**
* It is an extension of the traditional LSTMs. This can improve model performance on sequence classification problems
* is a sequence processing model that consists of two LSTMs

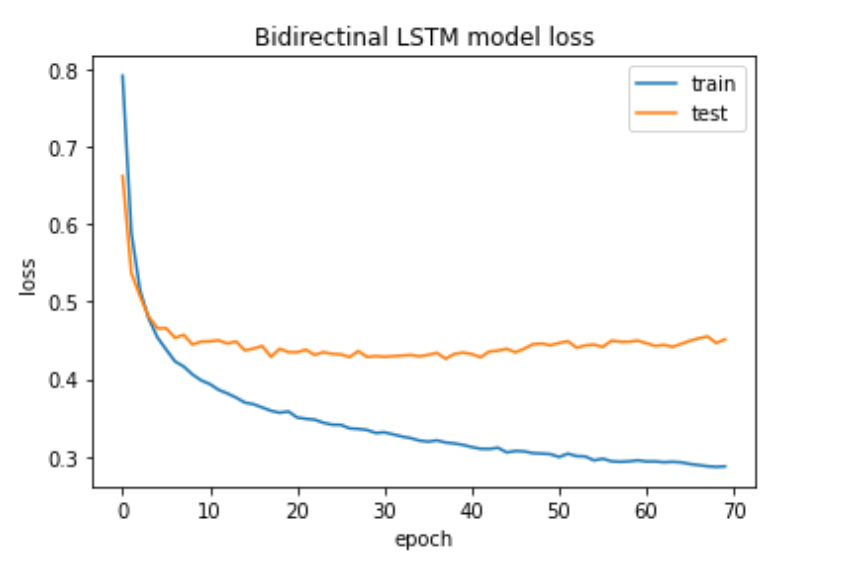
# 

BiDirectional LSTM Architecture for Sentiment Analysis

# 

# Fig 7: Bidirectional LSTM model Accuracy VS epoch

As we can see from the above graph, the model converges to 100% accuracy for the training set but for the test set it converges to 84%. Which means that this NN mode is not overfitted.



# Fig 8: Bidirectional LSTM model Loss VS epoch

# Similarly, the model converges to 0 for the loss of training set while for the testing set it converges to 0.45.

* + **Single LSTM Model**
* units or blocks that are part of a recurrent neural network and it is capable of handling long-term dependencies
* increases the memory of recurrent neural networks.

# 

Single LSTM Architecture for Sentiment Analysis

# 

Fig 9: Single LSTM model Accuracy VS epoch

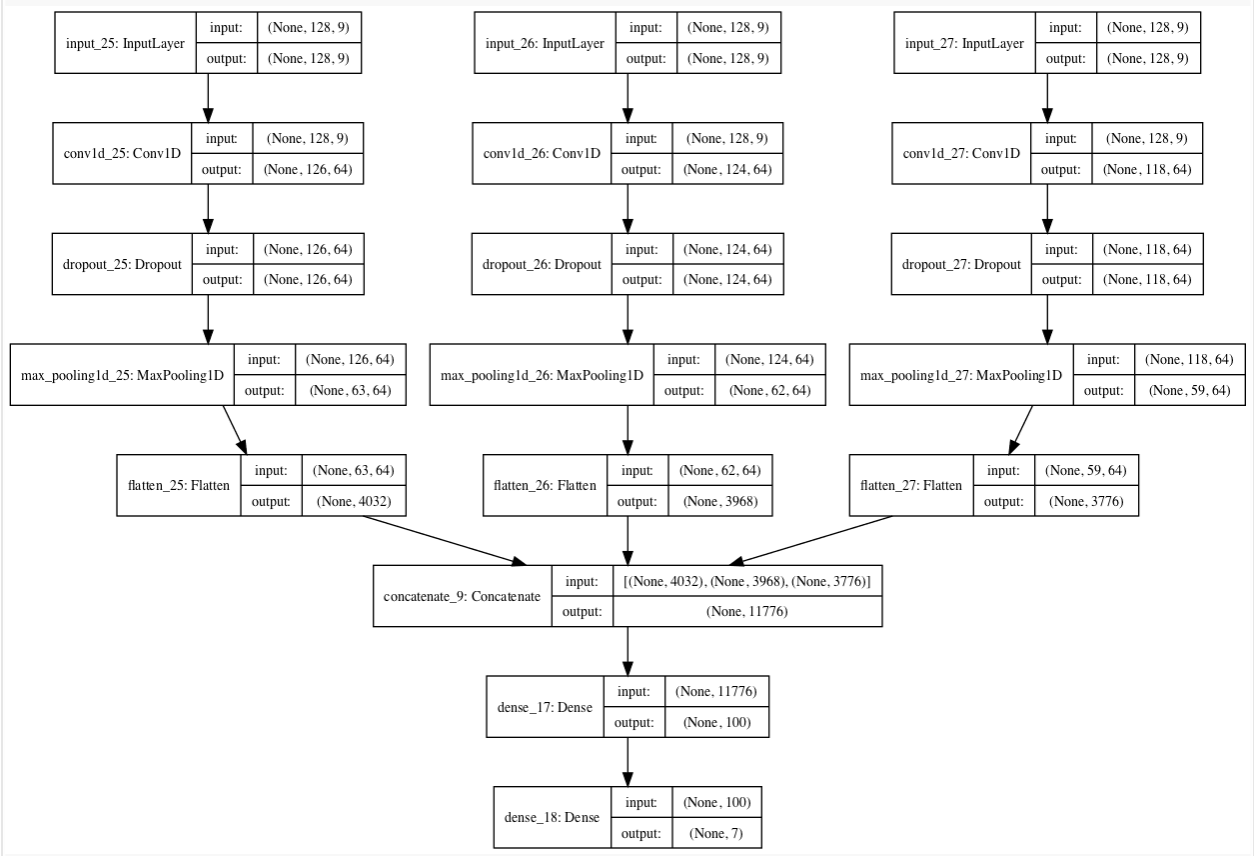
As we can see from the above graph, the model converges to 100% accuracy for the training set but for the test set it converges to 84%. Which means that this NN mode is not overfitted.

# 

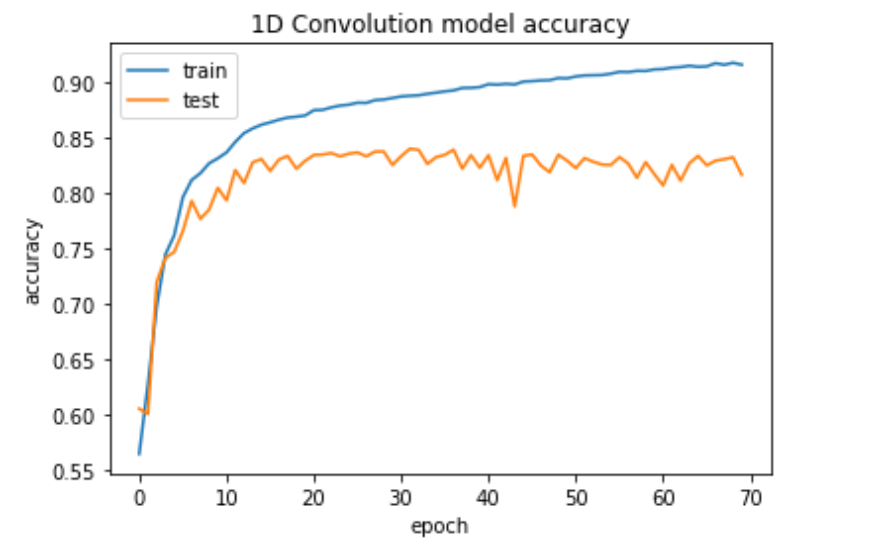
# Fig 10: Single LSTM model Loss VS epoch

Similarly, the model converges to 0 for the loss of training set while for the testing set it converges to 0.45.

* **1D Convolutional Model**
* is the simple application of a filter to an input that results in an activation
* The 1D Convolution block represents a layer that can be used to detect features in a vector.

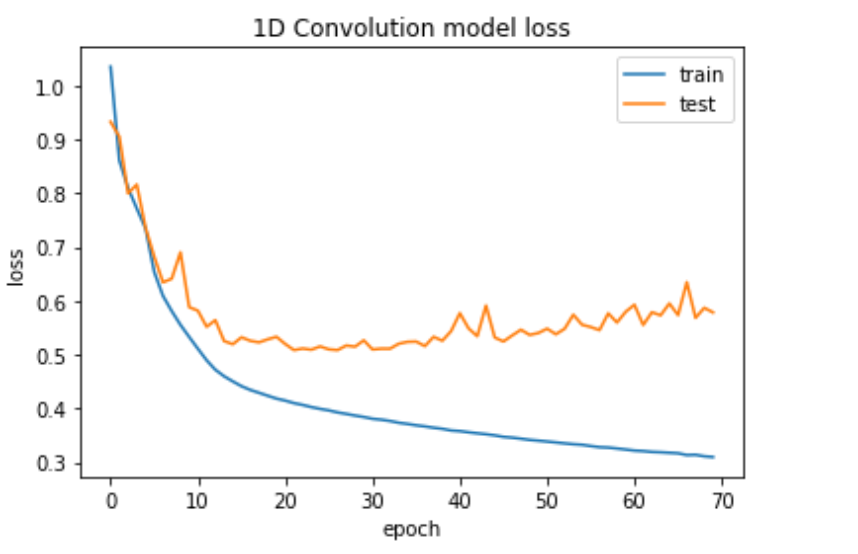


1D Convolutional Architecture for Sentiment Analysis



# Fig 11: 1D Convolutional model Accuracy VS epoch

As we can see from the above graph that the model converges to 100% accuracy for the training set but for the test set it converges to 81%. Which means that this NN mode is not overfitted.

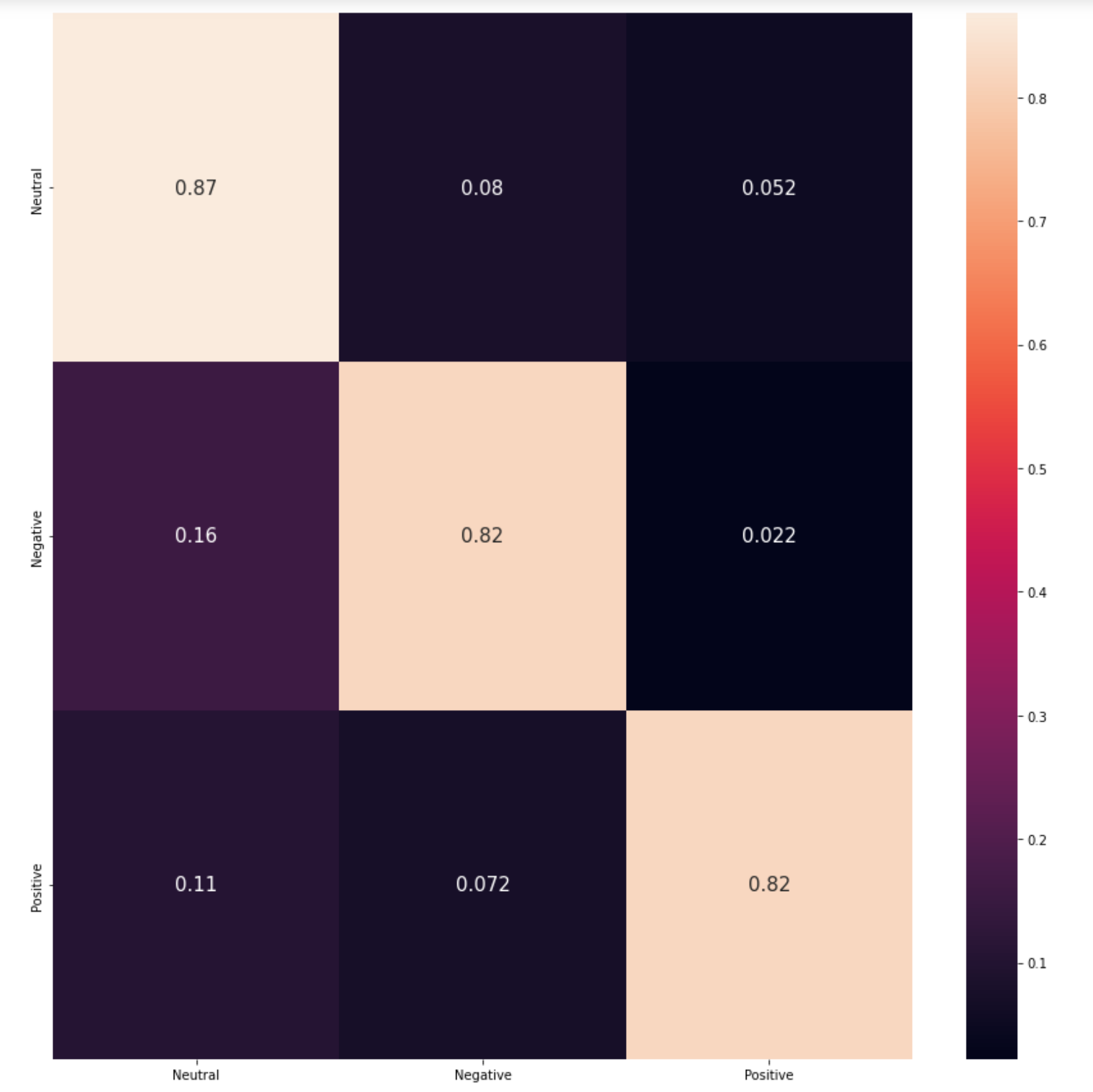


# Fig 12: 1D Convolutional model Loss VS epoch

Similarly, the model converges to 0 for the loss of training set while for the testing set it converges to 0.57.

**Confusion Matrix:**

We all know that accuracy is not a good metric to measure how well a model is. That is the reason why we will use the confusion matrix, that way we have a better understanding of its classification and generalization ability.

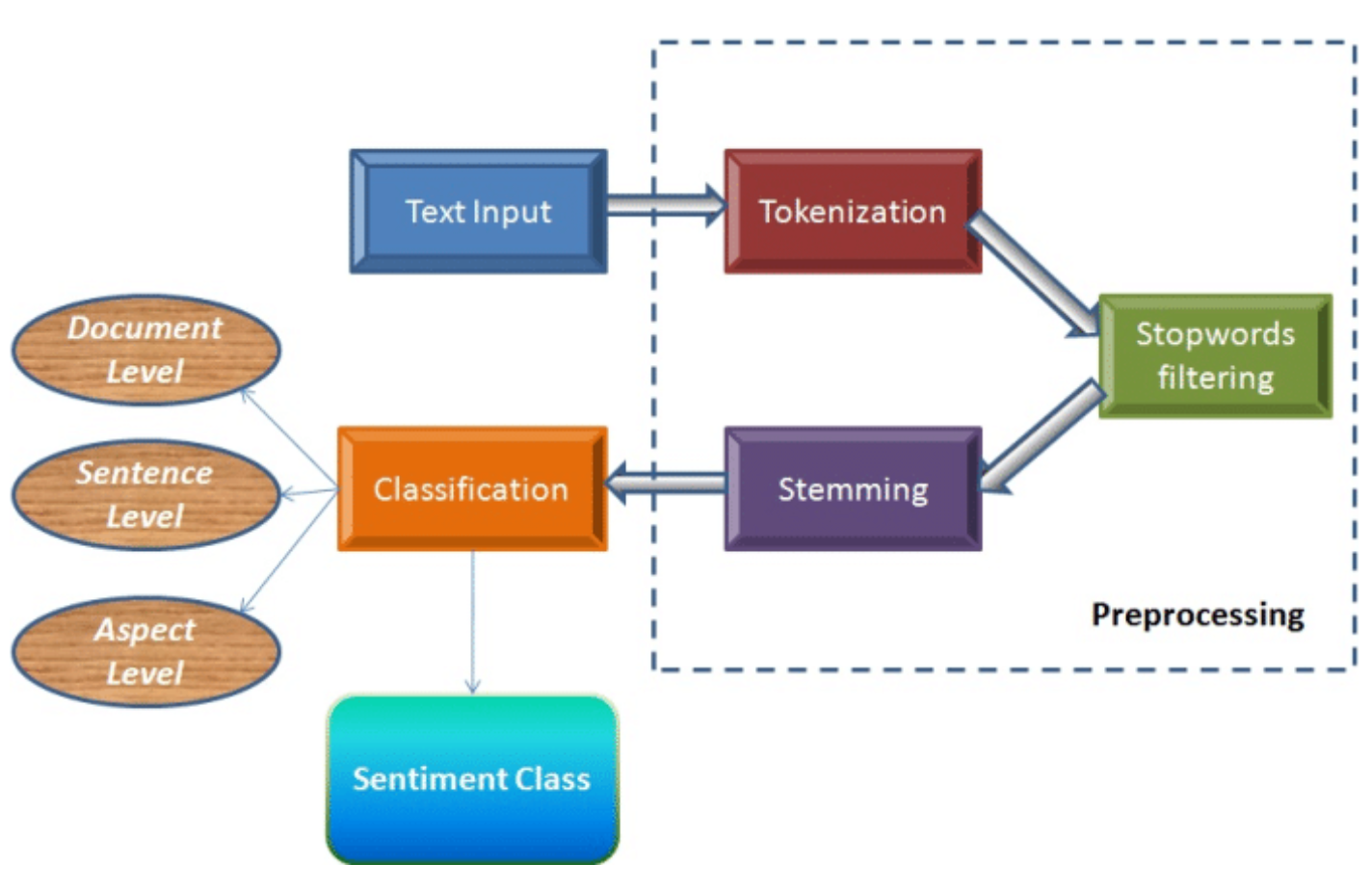
****

# Fig 13: Confusion Matrix for Bidirectional LSTM models

# 

Based on our findings from the dataset that contains 20,000 tweets from Twitter, we come to find that from the three Recurrent Neural Network Models which are the Bidirectional LSTM Model, Single LSTM Model, and 1D Convolutional Model, BidRNN gives us the best results. This can be seen when we check the val\_accuracy in the training logs. From this, we come to the conclusion that there will be no better score than the one that is achieved by BidRNN.

When looking at the training set accuracies of the three RNN modes, all of them converge to 100% accuracy, However, when looking at the accuracies for the test set, for both Bidirectional LSTM Model and Single LSTM Model, it converges to 84%, however, for the 1D Convolutional Model, it converges to 81%.

****

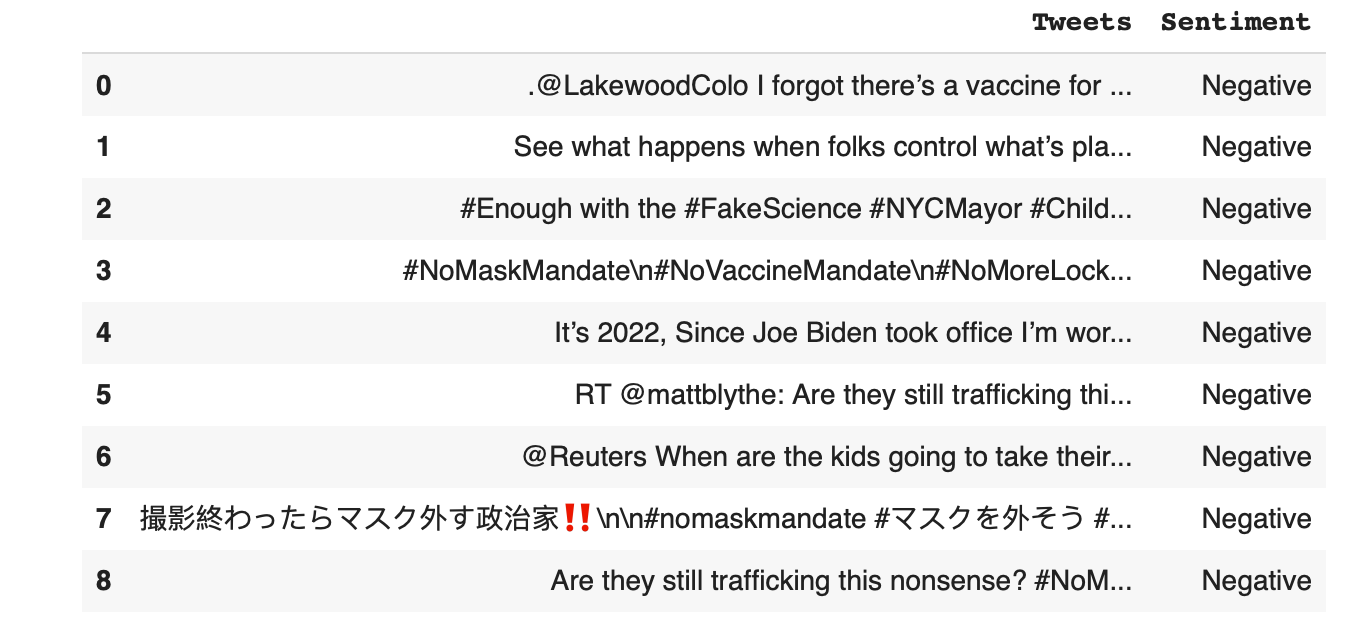
# 

Using the above finding, then we applied our best model, that is BidRNN, in predicting the sentiment of the people over the no mask mandate. For this first, we used Twitter's REST API to get all the tweets. The, we filtered each tweet by certain hashtags such as “#nomaskmandate'' and dropped the tweet if it did not have those hashtags. Then, we labeled each tweet by applying the preprocessing steps mentioned above. After the tweets were labeled, then we found the sentiment of each of the tweets.

# 

# 5 Results

The extraction of the thousand recent Twitter tweets led us to discover that the majority of people do not like the lifting of the mask mandate policy previously set by NYC. By filtering out the tweets by the hashtag, #nomaskmandate, we have shown a few below. Then, we ran the sentiment model and the result was that all of the tweets gave us back a negative sentiment. As seen below, we see negative tweets saying stuff such as “enough with the fake science” and “are they still trafficking this nonsense.”



# 

# 6 **Conclusion**

In conclusion, by using different Machine Learning models, we are able to see how people view the ongoing issue of the mask mandate. Many companies and businesses can use this information to set rules and regulations based on the overall feeling of the people around how they feel about the mask mandate. When dealing with a large set of Twitter tweets, companies and businesses can use the machine learning algorithms such as Bidirectional LSTM Model, Single LSTM Model, and 1D Convolutional Model to compare and check which one provides the best accuracy. With the implementation of the Twitter sentiment analysis model, the issue around identifying the sentiment of tweets can be solved. By finding the valency around the mask mandate policy, a future component that can be tested around is how people from upstate vs NYC feel about the mask mandate policy.

**7 References**

[1] Vishal A. Kharde & S.S. Sonawee, “Sentiment Analysis of Twitter Data: A Survey of Techniques.” Pune Institute of Computer Technology, Pune University of Pune (India).

[2] Muhammad Umer, Imran Ashraf, Arif Mehmood, Saru Kumari, Saleem Ullah, Gyu Sang Choi, “Sentiment analysis of tweets using a unified convolutional neural network long short term memory network model.” 2020.

[3] Omar Y. Adwan, Marwan Al-Tawil, Ammar M. Huneiti, Razan H. Al-Dibsi, Rawan A. Shahin, Abeer A. Abu Zayed, “Twitter Sentiment Analysis Approaches: A Survey.” University of Jordan, Amman, Jordan, 2020.

[4] Efthymios Kouloumpis, Thersea Wilson, Johanna Moore, “Twitter Sentiment Analysis: The Good the Bad and the OMG!” University of Edinburgh, 2011.

[5] “Embedded Value in Bloomberg News & Social Sentiment Data.” 2016.

[6] Haya Fathim, “Sentiment Analysis of News Tweets.” Rochester Institute of Technology, 2021.