Chapter - 6

Twitter Sentiment Analysis of Covid-19 Vaccination Using Deep Learning

Varsha Naika [1], Dr. Rajeswari Kannanb [2], Snehalraj Chugha [3], Ahbaz Memona [4], Himanshu Chaudharia [5]

[1,3,4,5] MIT-WPU, Dr. Vishwanath Karad's MIT World Peace University, Pune, India.
[2] PCCoE, Pimpri Chinchwad College of Engineering, Pune, India.
Email: [1] varsha.powar@mitwpu.edu.in, [2] kannan.rajeswari@pccoepune.org, [3] snehalchugh2016@gmail.com, [4] ahbazmemon0@gmail.com, [5] himanshuchaudhari2346@gmail.com

Abstract— Covid-19 had consequential social, economic, and extreme mental outcomes on the community, where media platforms like Twitter increasingly became essential networking mediums generating information with a large volume of reports, views, opinions, and information shared by individuals and authorized outlets. In 2021, when the second wave of COVID emerged in India, we recognised the fastest outbreak with more than 20 lakh cases in April's first half. Until then, India distributed over one billion vaccine units with two producers: Bharat Biotech, producing Covaxin, and Covishield, OxfordAstraZeneca's vaccine, by SII (Serum Institute of India).

We collected datasets for analysis, and applied our novel algorithms for preprocessing, i.e., removal of URLs, @, #, contracted words, punctuations, numbers, POS, etc. Converted tweets into tokenized words, used stemming & lemmatizations, then applied neural spellchecker. Using our in-house algorithm, we cleaned around 500 tweets in just 0.5 seconds, getting rid of duplicate and redundant tweets. A word cloud with classes: Positive, Negative, and Neutral was constructed which then used neural network to predict them, resulting in 97% training and 99% testing accuracy. Results aid in improved policy design, keeping citizens' perspectives in mind, and an aware government about issues like vaccination shortages, food, poverty, etc.

Keywords— Covid-19; Pandemic; Sentimental Analysis; Word Cloud; K-Means Clustering; Natural Language Processing; Vaccinations; Twitter; Covaxin; Covishield.

I. INTRODUCTION

In almost every country, the new coronavirus is spreading rapidly (COVID-19). Thousands of citizens have been infected, and huge numbers have died from the illness worldwide. Twitter activity has also increased by roughly 25% within the same period.

It used to be widely accepted that the virus was not infectious until the beginning of January 2020. However, researchers later identified it to be the new coronavirus as the source of the sickness and found that it could transmit from individual to individual. Subsequently, the city of Wuhan [1], a metropolis region with 11 million inhabitants, was ordered to remain closed down, and the province of Hubei swiftly followed suit. The illness eventually led numerous Chinese regions to be placed in quarantine in February. To limit the radiation, China halted its commerce from February to March of 2009 [2], [3]. Wuhan, a metropolitan region with 11 million citizens, was soon put on lockdown with severe orders for everyone to stay inside, and the state of Hubei quickly fell under the lockdown. This subsequently triggered the implementation of lockdowns in several Chinese regions in February. China needed to stop the economy from running for the majority of February and the first two months of March to limit the spread [4]

China's recent efforts to control the epidemic since late January 2020, while preventing the spread of the disease to other countries, has only succeeded in spreading the sickness throughout the world [2]. As every evidence proves, the virus is naturally occurring and has originated from bats or may have arisen through an intermediary mammal species. However, transferability between individual to individual is uncertain. In particular, from animals and humans, the transferability is unknown as well yet. This worldwide catastrophe sparked an epidemic proclamation by WHO on the 11th of March [5], and several national emergencies followed. Using physical distance (including school closures, nightclubs, eateries, cinemas, and encouraging companies to only have their executives work from home). It's indeed strongly discouraged or prohibited to even have major public assemblies such as concerts, graduation ceremonies, and sports activities. It is said that the economic effect of mitigation has decimated countless companies, but according to reports, over 40 million individuals in India have applied for initial unemployment benefits [6].

Varsha Naika – "Twitter Sentiment Analysis of Covid-19 Vaccination Using Deep Learning" Pg no: 75 – 92.
https://dx.doi.org/10.36647/MLAIDA/2022.12.B1.Ch006

Individuals utilize the media to learn more about their personal health decisions. Because of the amount of data available [7], this may be even more relevant in the case of the COVID-19 outbreak. Despite the fact that fresh information is always flowing in, the primary questions of viral transmission, post-recovery antibodies, and medication therapy remain unanswered [6], [8], [9]. In light of the increasing amount of information, many people resort to social media to get clarity. Several studies have found that vaccination material is extensively distributed throughout social networking sites, with particular attention paid to how it is depicted on the Internet. Social media conversations around vaccines have grown following current occurrences in the news. Using platforms that allow people to discuss issues around vaccination, content appears to show up throughout individuals who have similar attitudes about vaccinations [8], but hardly ever among those with opposing beliefs.

To promote vaccination on social media [2], [9], the public health service may be prevented from increasing its spread by ideological isolation. Much research has gathered Twitter data following the COVID-19 epidemic to help comprehend public responses & debates concerning COVID-19. The volume of anti-vaccine material disseminated throughout social media is impressive. The current research, although early, shows that exposure to this type of information may impact vaccination attitudes and, as a result, vaccination delay [10]. Confusion and misinformation are both spreading quickly as individuals try to comprehend the best way to defend themselves, their loved ones, and also to post as many provocative comments as possible, which hinders one's reading comprehension.

Phase one of the vaccine campaign focused on making sure that 30 million healthcare providers and 2.7 billion priority population members were aware of the immunization opportunities [10]. It was anticipated to be finished by July [11]. Although the Indian government has recently launched two vaccines for the nation's huge campaign, the Ministry of Health has revealed that even though the vaccines are in high demand and logistical issues will make it difficult for individual people to pick and then choose the antibiotics, they will have to go through with the government's decision. Although everyone just above the age of eighteen would be entitled to receive COVID-19 vaccinations in India, which is something the federal government stated on Monday, May 1st [9], [10], [12] not everyone will receive vaccinations. New expenditures and programs relating to immunizations for the citizens in India also were announced. Since getting relevant, reliable, and high news from these sites means that there will be precise and up-to-date data upon the COVID-19 epidemic and the vaccine review for people in that age group bracket of 18 to 45, this will help keep the pandemic at bay. Some Indians can't be vaccinated as they wait, and thus they're also using Twitter to voice their views.

The goal of this research is to analyze the feelings and attitudes through Twitter of the Indian people towards both Covishield and Covaxin vaccines for individuals aged 18 and above, which was authorized since [4], [11]. The public social media data published by people worldwide will be utilized to discover the major ideas, beliefs, emotions, and subject matter that individuals have around the COVID-19 pandemic vaccines. Information such as this can assist public politicians, healthcare providers, and citizens in identifying important concerns and offering better solutions.

II. DATA GATHERING

This research intends to explore the public discussion and sentiments linked to the COVID-19 spread by analysing tweets gathered with the use of Tweepy Python package for leveraging Twitter's Streaming API [12], [13] in order to extend the knowledge on public responses. Twitter specifies the language of each tweet when using its streaming API. Not surprisingly, given worldwide Twitter usage, the majority of tweets (57.1%) are in the English language thus we focus on collecting only English language tweets.

We started by analyzing the number of average tweets we could retrieve, thus we started collecting tweets from February 25, 2021, to April 26, 2021 across India except Jammu & Kashmir, as we had received the least tweets from that State, given in the Figure 1. While the approach proposed in this paper can be extended to adding parameters and searching for a variety of tweets, in the present study all the tweets without any filters have been retrieved, approximately retrieving in a total of about 2,47,627 tweets. The number of tweets that have been daily retrieved has been plotted in the above graph. This helped in understanding the average amount of tweets that are taken daily, which was giving a threshold value of 4128 tweets on average, Figure 1.

As in our research, we are aiming to perform an analysis of the sentiments of people around the globe related to COVID [14]. The vaccinations were granted to them after May 1, 2021. We mapped out to prepare a dataset of 50,000 tweets, thus understanding the average tweets we get daily (4128) it took us approximately 13 days to download and retrieve. As an allowance of vaccination was provided and there was a hike of citizens reacting with a variety of emotions, it was perfect to perform the retrieval of tweets from May 3, 2021, to May 16, 2021, Figure 1. In conclusion, we acquired a dataset of around 49,345 rows of tweets filled with hashtags and about a variety of topics.

Furthermore, we then created an algorithm that performs a keyword & hashtag extraction over a certain tweet, this helped us in retrieving a particular number of tweets just related to our topic. We noticed that there were in total approximately 2,657 unique hashtags and keywords on general topics, used in all the tweets in our dataset. After understanding our topic elaborately through various websites, we created a list of 724 keywords out of 2,657 hashtags related to COVID-19 and VACCINATION. We then scanned the corpus with our skewed keyword list, obtaining a set of 15,174 related to our problem statement tweets by

people across the globe. These filtered tweets were further sorted in time-ascending order and converted to a bag-of-word representation. All analyses were performed using Tweepy & Python (3.7.0.0)

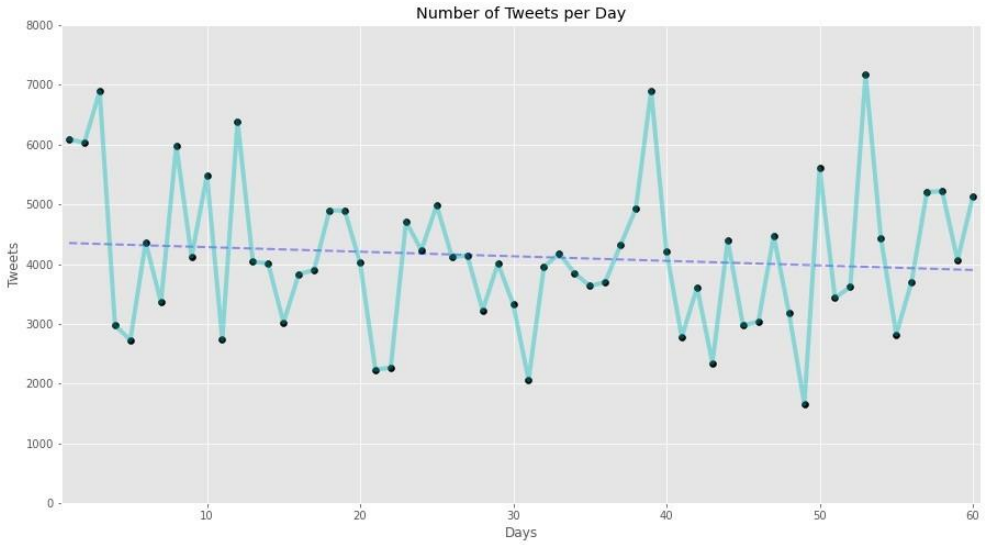

Figure 1. A threshold quantity of tweets streamed over a period of 60 days, starting on February 25, 2021, and ending on April 26, 2021, using that information gives us an approximate daily stream of tweets.

III. DATA DESCRIPTION

From March 3, 2021, to March 16, 2021, these tweets were collected from around the globe. The data stream we retrieved had 15174 rows and 10 columns, with the majority of them being categorical and objects, Figure 2. For our suggested statement, all of them must be categorical. This dataset has 10 attributes: 'Tweets,' 'User,' 'User statuses count,' 'user followers,' 'User location,' 'User verified,' 'fav count,' 'rt count,' and 'tweet date.' Tweets are text strings that include emoticons, hashtags, usernames, and other characteristics. User is the username, User Status Count is the number of tweets issued by the user, User Followers is the number of followers each user has, and User Location is the user's location from where the tweet was tweeted. However, since the majority of Twitter users do not allow geotagging for tweets, the location was retrieved from the profile of the tweet's username using the Tweepy Python module. They are from different places and not from a particular location, having in total about 3240 unique locations. User Verified refers to a user's verification, which is a blue tick that each user receives after their Twitter account has been verified. The number of times this tweet has been favorited is shown by the Fav Count. The retweet count indicates how many times the tweet was retweeted, while the tweet date displays the date and time the tweet was posted. As a result, the dataset includes 75% of the retweet count as 1. The total number of verified and non-verified individuals is 1486 and 13688, respectively.

IV. DATA PREPROCESSING

Preprocessing of the raw experimental data is critical since it improves the quality of the raw data. Bringing our tweet into a reliable and computationally efficient shape is simply referred to as preprocessing. To make the tweet more acceptable for machine learning and neural network algorithms, we should modify the tweet and preprocess it.

4.1 Preprocessing of Tweets

When we discovered the data, we had to cleanse it and ensure that it was free of repetitions or mistakes so this dataset had no error. We went ahead and completed this by utilizing our processing algorithms that were not just for cleaning data but for removing repetitive tweets as well. Many techniques are used in order to prepare the tweets for modeling sentiment analysis [13]. The system's success rate and runtime will both drop if data cleansing and removal of duplicates are not performed. The rise in the success rate occurs when you get rid of words and terms that are useless.

4.1.1 Cleaning of Tweets

Transformation of Cases and WhiteSpaces: For consistency, all capital letters are transformed into lowercase. Machine learning techniques are case sensitive and so, the words "Vaccination", "VACCINATION", "vaccination", and #vaccination is all rendered as "vaccination". To eliminate unnecessary spaces, we use strip () to completely remove all white spaces in the tweets.

Removal of URLs and hyperlinks: Many of our tweets contained URLs or hyperlinks which were not helpful either. Due

to the nature of links in tweets, they cannot be used for sentiment analysis, thus they must be deleted. A large number of scripting languages, online resources, and text-based patterned expressions tools make use of regular expressions. They are widely employed in statistical analysis, data analysis, the transformation of data, and in a wide variety of other data-cleaning and data-transformative processes. Data cleaning is a critical procedure that should be completed for the machine to enhance learning [15].

Regular expressions were used to remove the URL (.org, HTTPS, .com) and links from the tweets. When doing this operation, we encountered a snag: the algorithm was unable to offer us 100 percent accuracy in the removal of hyperlinks and URLs from the dataset string whenever executed. For example, when the phrase "HTTP(s)" was removed, the letter "s" in the phrase remained. We streamlined the process by developing an efficient function that eliminated all the unnecessary URLs, and so created an ideal outcome. It did this with the least amount of effort while delivering excellent accuracy.

Removal of Usernames & Hashtags: The hashtag "#" makes all the Tweets to which it is related readily available. It acts as an indicator that a piece of information about a certain topic or categorized under a certain topic could be provided [15]. A user's profile is marked by the @ symbol at the beginning of a handle. For example, if tags such as @harbour or #vaccine, remain in the tweets when there is no data cleaning performed then this might lead to model inaccuracy and lower the success rate. Instead of repeating the elimination step using Regular Expressions, we developed an optimized function for this as well to take advantage of irregularities and maximize outcomes.

Removal of Contraction of words & Punctuations: Truncated forms of sentences or words are called contractions. They are often encountered in English, whether that be in writing or verbal. For many words, we omit the vowels while forming the contractions. Eliminating contractions helps in standardizing texts and is beneficial when dealing with Twitter data. It also has the added benefit of improving sentiment analysis by making sure the words don't contradict one other. Thus, looking at words like "I'm," "Could've," "Wouldn't," and so on that have punctuation, which can't be completely erased or deleted since the words' meanings could get affected.

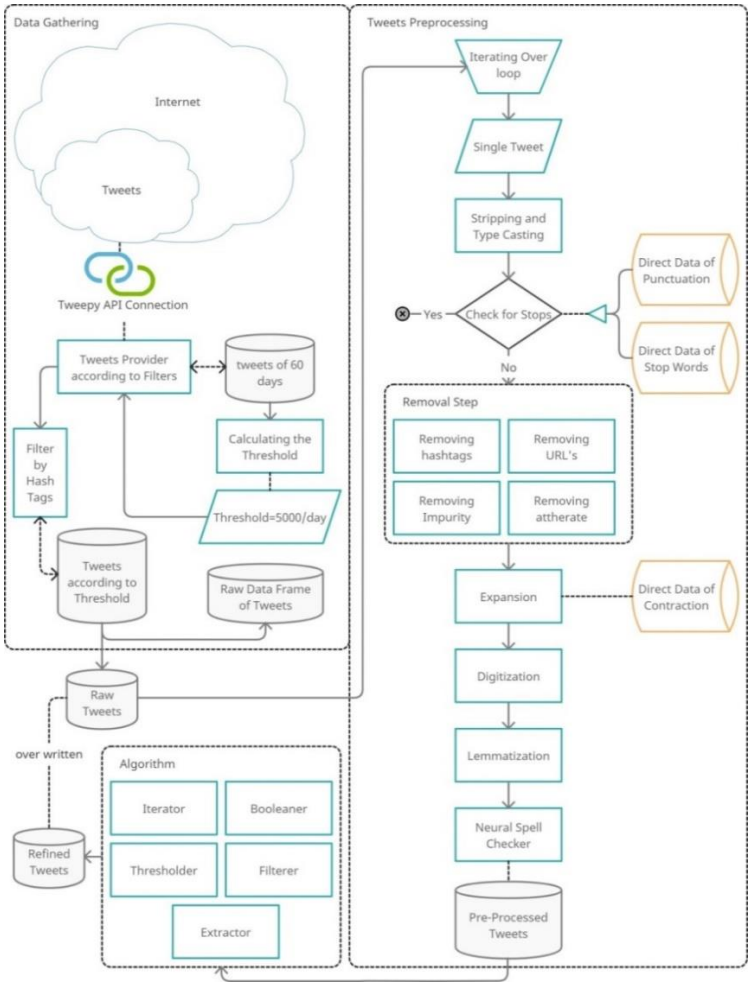

Figure 2. The diagram above depicts all of the processes involved in collecting data, preprocessing tweets, passing them through the algorithm, and resulting in refined tweets. All these tweets are overwritten after the processes in refined tweet

The library provides “Contractions” as a variety of terms and made it possible for the user to input more words, making it suitable for our use case. Therefore, we utilized this library, but in addition, we built our dictionary of words. The greatest pleasure here is that the library, as a result of a multitude of additions, was able to carry out the straightforward process of replacing the contractions without making the search or replacement process longer. Machine processing is hindered by the punctuation characters, such as, ‘!’, ‘~’, ‘’, and ‘?’, and thus these characters are eliminated as well.

Converting numbers to string values: Additionally, we also observed the presence of fractions, whole numbers, and decimals, all of which supported our overall problem statement. To better match our results, we transformed the figures to words using the num2words package. To provide an example, 16, One Six, sixteen, etc. gets unanimously converted to sixteen.

Tokenization: The process of tokenization divides tweets into a list of words. This approach separates words, sentences, and other linguistic units from the text into pieces called the tokens. This is a procedure where we systematically work our way through each tokenized phrase and filter words whose length is less or more than a set threshold. We decided to utilize the word_tokenize function from NLTK [12]. Twitter-specific tokenization provides a better way of verifying that URLs and hashtags are completely separated in the tweet.

Removal of Stop Words: Certain terms are abundant in tweets, such as, ’and’, ’a’, and so on. We may focus on the relevant aspects of our content by eliminating unnecessary words. By removing stop-words, we significantly minimize the clutter in tweets, which is unnecessary for issue analytics. We utilized the library called stop-words present in NLTK to clean these tweets and it aided in removing words such as the, an. with, at, etc. [16].

POS Parts of Speech: This approach can identify nouns, adverbs, adjectives, subjects, and objects in a phrase, and it further categorizes the part of speech of an individual word to aid in analyzing sentence construction. It is additionally used to determine the word’s raw form of connotation disambiguation. A large number of features are available by completing this phase. By doing this step, the words representing the structure of the phrase and the content of the message in the field to which the sentence belongs can be obtained. To build an approach and also to get word meaning for just tweet tags, a technique created using the NLTK class called POS tagger was utilized [17]. To accurately identify the sentiment score of a phrase, you must first arrange the sentence according to part of speech. The accuracy of attributing emotion polarity to sentences will be impaired as consequence, meaning favorable, unfavorable, or neutral.

Stemming and Lemmatization: In the normalization process, words are stemmed and lemmatized to create a normalized version of the term in the text. Word stemming is a methodology that recovers the word’s basis in the sentence. A novel manner to normalize a word is to remove its suffix from the term. To accurately determine sentiment, just stemming words with a length larger than two (“a”, “is”, “an”, and “the”) are used, since words like “an”, “the”, “is”, and “are” are not included into the sentiment vocabulary while measuring the word’s polarities.

The word lemmatization technique transforms affixation and/or changes a vowel from either the base or dictionary form of a word. A lemma is the result of the process of word making. The lemmatized word is the entrance to the WordNet since the lemmas are the words that already have a core definition of the phrase that is sought. Computing lemmatization using an approach thus creates a lemma, which is then transferred to WordNet dictionaries to obtain a new sentiment for the term. Using the WordNetLemmatizer classes (accessible via the WordNet stem package) within the Python NLTK stem bundle, one may conduct lemmatization in terms of the corresponding character-by-character [12]. The use of WordNetLemmatizer produces the lemma (root meaning) of such input sentences while ensuring accurate representation. For example, “Vaccines” is a variant of the root word “Vaccine”, that is defined throughout the WordNet vocabulary.

Preprocessing is essential as it influences the accuracy of models of learning. Figure 3 illustrates the same thematic of our approach. We can see here that the time required to clean from version 1 to version 3 is enormous. Our versions have been improved during the course to ensure that it required as little time as feasible. For the model to learn effectively, data cleaning is an essential step. The model will thus get a better result because of the lack of redundant words and phrases in the data set. Now, we start by reviewing each function we used to clean our dataset, detailing how our accuracy and time to complete each step were polished to their highest possible quality for each iteration, getting to version 3.

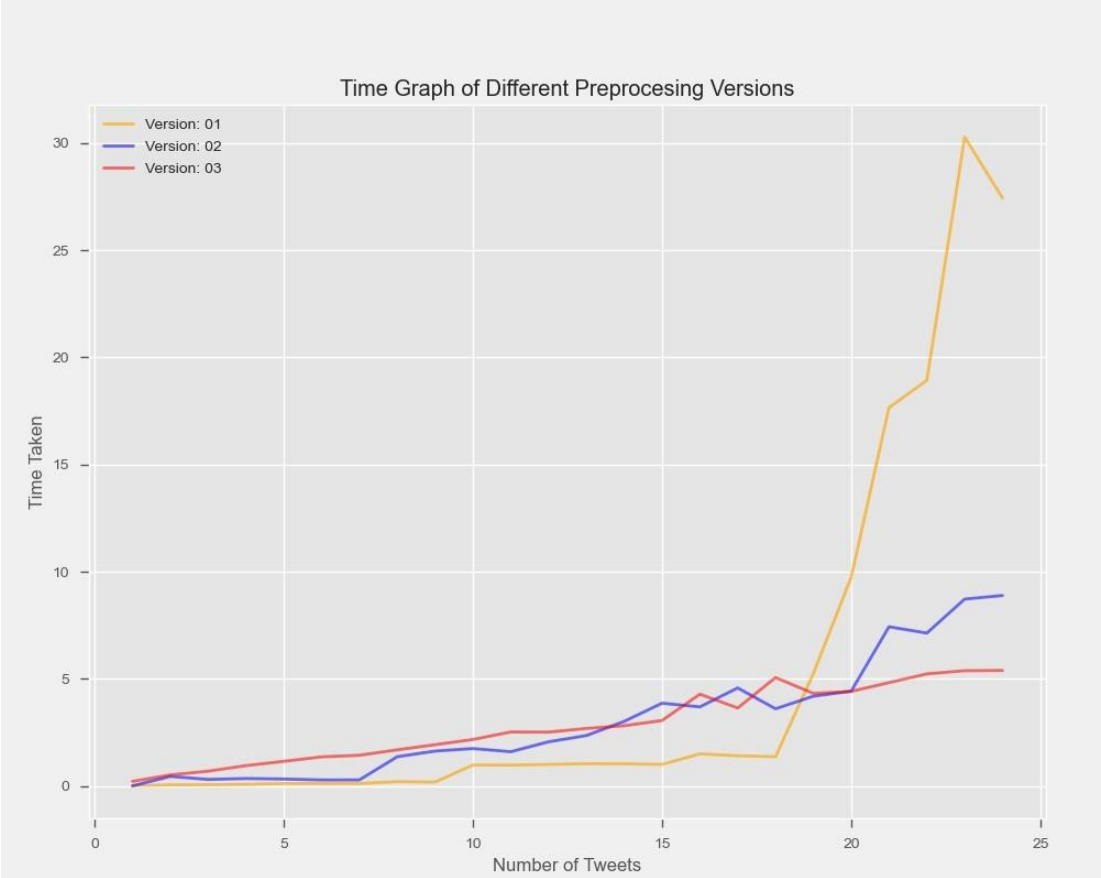

Figure 3. The following graph is a comparison of three different versions throughout time, the number of tweets where the most optimum results were received in our final version 3.

4.1.2 Algorithm

Every Twitter user has access to the features of the user's tweet, like retweet and reply count. A tweet may be described as the ability to affect someone emotionally, with an opinion, or through influencing their conduct. For Twitter, a favorite is an indicator of reader approval for a tweet. Tweeting and retweeting indicate that the user agrees with the sentiment expressed in the tweet and is eager to share it with his followers. Commenting (re-tweeting) and disseminating further the opinion of a tweet demonstrates that the reader wants to discuss and share their views.

However, people may settle on merely copying someone's tweet and altering it slightly to make it distinct. According to this approach, there were many, many tweets identified similarly as shown in Figure 4. Specifically, the correlation between the support for a tweet and the prediction affects the prediction, resulting in an incorrect prediction. Because there were so many redundant tweets in the dataset, we built an algorithm to prune these posts, so we could make more accurate prediction.

When we started, we developed an algorithm that examines a tweet and runs a set of operations to generate a dataset of unique tweets, which also comprises the least-repeated terms and each tweet is unique inside the dataset. Using the algorithm, we found several areas to modify since we didn't get the precision we anticipated in the removal. Everything is detailed down to the smallest possible detail in five different versions, and all of these versions included numerous modifications which enabled us to achieve our objective of an optimal level of accuracy when we finally arrived at our final design, version 5 Figure 7. Below we have explained how we began with our first version and with incremental improvements culminated in our objective.

So, we started by implementing the iterator function in our algorithm, which locates all of the tweets in the dataset and afterward compares each of them to all of the other tweets in the dataset. It takes the tweets and then tokenizes both of them, i.e., the one which is comparing itself with all the tweets and the other one which is being compared. When we tokenize both of these tweets, we pass them via another function called boolenear. The function boolenear takes each tokenized word and checks each word in the tweet to see whether it matches any word in the other tweet. Following all the iterations, the algorithm produces a list of booleans that imply whether the first word, second word, third word, and so on of the tweet are all equivalent to the first word, second word, third word, and so on of the other tweets.

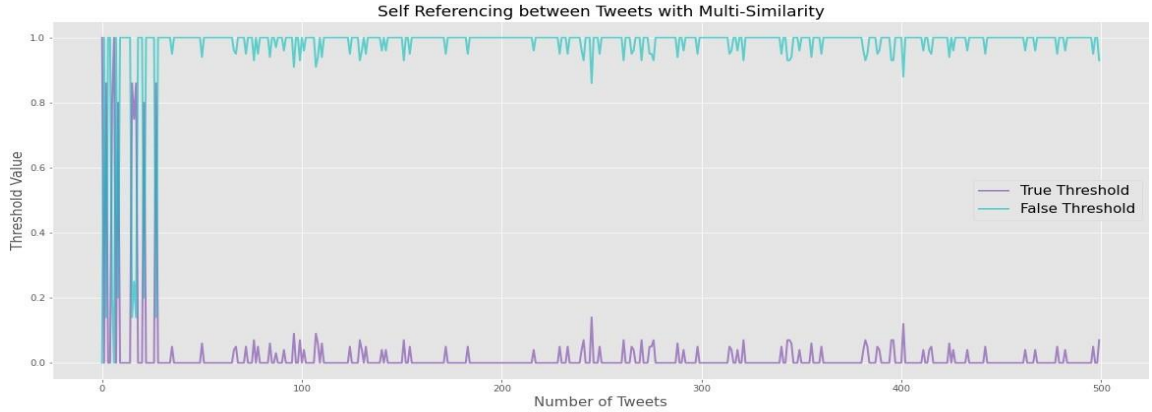

Figure 4. This graph shows the location of words in two tweets being similar at places before cleaning, here tweets are compared and if two tweets included the same words in the same locations, those tweets had to be cleaned.

For each place in the tweet, if the words are comparable, it will return True or False declaring true, or else marking false if the phrases on the same spot are different. The most essential aspect to consider when there is a comparison in tweets is if the length of both tweets is comparable. For that reason, knowing that not every tweet would be comparable in length, we set the boolenear to disallow comparisons for the additional words and mark those false.

The reason we required these boolean values was to assess the percentage of how much the words were similar in two tweets. To assist in comprehending the list of true and false, we applied the Thresholder algorithm to our dataset, which computed the percentage of how much each tweet in the dataset was similar to each other. Thresholder operated by determining the amount of true and false values for a given set of compared tweets, determining which are unique, and setting an exclusion level for the set.

After the percentage is calculated from the Thresholder, it's passed on to a function called filterer, which uses it to filter tweets. This function analyzes the percentage and verifies that the false value threshold is less than 25% for false and also greater than 75% for true, thus implying that the filter has indicated that the tweet is similar. In turn, if the condition that holds are satisfied, then we understand that those two tweets aren't similar, and therefore they should remain unchanged in the dataset, Figures 4 and 5.

Our dataset was almost ready to be cleaned, but the elimination of duplicate and repeated tweets remained to be done last. But if we eliminate all these tweets that the filterer was looking for, then the dataset may become empty. Therefore, we construct an Extractor function that takes the tweets, creates an array that includes all the boolean values, namely true or false for each word of the tweet which each Tweet compared with the other Tweets existing in the dataset earlier. This function searches for the duplicity in the dataset after comparing all the tweets in the array. Where a set of similar tweets is found, just one tweet with the most words saved while the others, that are partially duplicates, are therefore eliminated Figures 4 and 5.

Once our tweets had been filtered, where all the above changes were made in our versions 1-4 Figure 7, We discovered an issue with our system. Due to several iterations and an inordinate amount of repetition in the reading and comparing of tweets, they required a lot of time to process. Tweets used to go through certain important phases while they were iterating and comparing with themselves; but rather just the extra time invested in self-comparison was useless. So instead of having 6 individual steps working step-by-step, we implemented an algorithm consisting of 2 phases that utilize time efficiently giving us better and optimum results, which we implemented for our version 5, Figure 7.

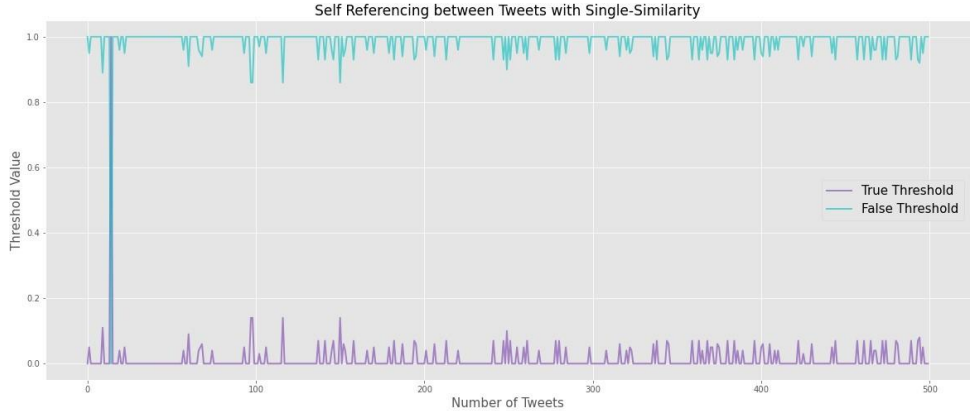

Figure 5. This illustration illustrates a comparison of two tweets after cleaning and here, no duplicates were found in either of them.

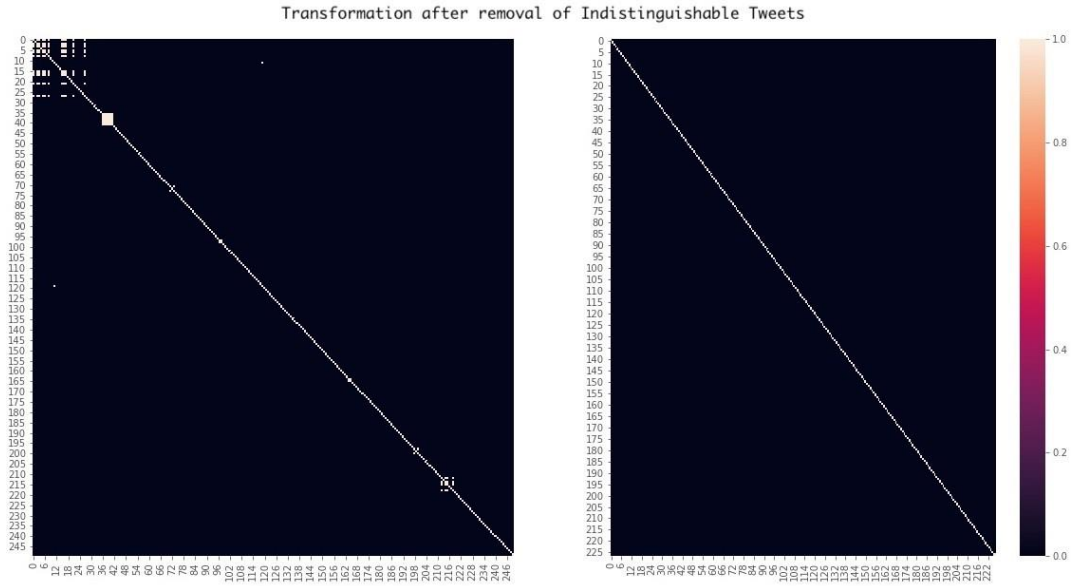

Figure 6. Our 1st algorithm is noticeably different from the 5th's. The diagonal line shows the self-referencing present in the graph on the left side with duplicates. Once we'd gone through the first step of our final algorithm, the distortions were no longer there which are shown in the graph at right side.

So, in Phase 1 of the algorithm's version 5, we start by taking a NumPy array of size n*n but being only 1 dimensional, instead of using for a list, Figure 6. The iteration would have been slow if we'd have picked Cuda, Data-frame, List, etc., so we opted for one dimension, which resulted in fast and efficient computation. In the previous approach, we used the sum function over each tweet, one by one, to aggregate the data. We replaced it with an incrementation checker, this eliminated tweets that just weren't helpful and therefore only kept the one that we required for processing, decreasing the time necessary at a higher level.

We did some additional tweaking to the filter algorithm in Phase 2, Figure 6, Which was recalculating true and false for each word in the tweet differently, so when examined more closely, we saw that none of it was necessary, thus we only noted one of the two, i.e., the percentage of the 'true' in tweets and allowed false to be computed by subtracting the percentage of true from '1'. We also reduced the portion of the algorithm where tweets in the array were repeatedly compared with themselves, which increased the algorithm's performance exponentially. Additionally, we just had to write two lines of code to produce the same threshold value rather than the long code that we used in the prior approach.

This is how adjusting minute aspects of functions since our initial algorithm's version 1, all of it boosted and helped us achieve our goal of producing cleaner and more uniform tweets in the least amount of time, Figure 7. preprocessed with the version 5 of our algorithm.

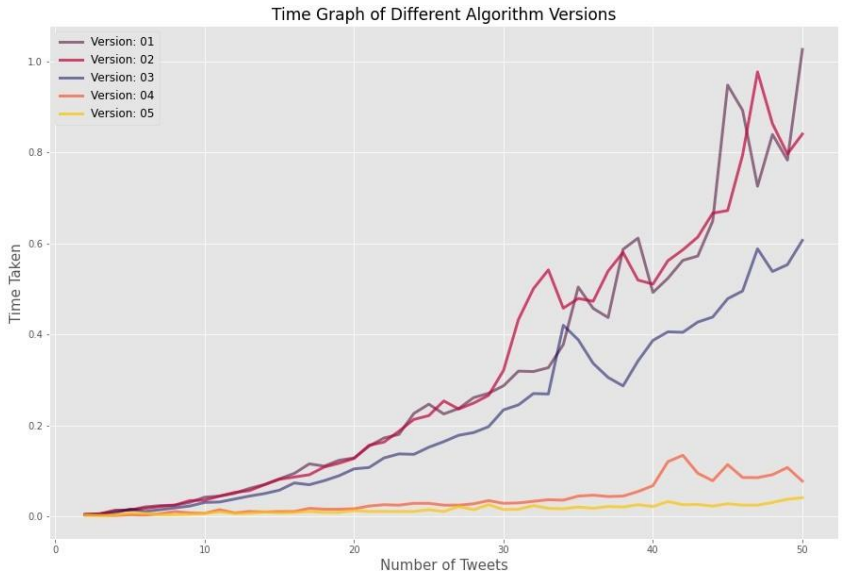

Figure 7. This graph illustrates the relative performance of all the algorithm versions on a certain number of tweets. We have concluded that the version 5 algorithm is optimal and quick, when compared to each of the other different versions.

4.2 Preprocessing of Data Frame

We ran our preprocessing tool and were left with about 11129 tweets, Figure 6. We tried to learn about the numbers of users, unique users in the dataset, the most frequent user, users who are verified, the number of retweets they've received, and their follower count in this processed tweet dataset. The preprocessing stage was so effective that the duplicates had already been eliminated.

Furthermore, we found that the tweets which were tweeted by lots of users were sourced, copied, framed, and added, etc. from various accounts, with verified ones being mostly sourced. Thus, being left out with no unique information in most of the non-verified tweets. There were many tweets that merely contained zero retweets, no location, etc. Also, the accounts themselves had almost no reach. Thus, only tweets with high reliability were included in the dataset; and to get these tweets, the dataset was filtered over the various filters given below: -

- User's account is verified, because they receive a higher reach and mostly are the highest active users, Figure 8.
- Tweets having retweets count greater than zero, as shown in Figure 9, they have higher reach and comprise around 33% of our dataset.
- Adding those tweets whose users, in particular, had following greater than 50 & also the location is null.
- We appended all the tweets with location not null, which is shown in Figure 10, due to their activeness in various regions.

These various filters allowed us to get relevant and particular tweets that had been narrowed down out of the rest of the Twitter flow in relation to our problem statement. Once the algorithms were completed, we had around 9525 Tweets remaining in the dataset on which we then run the algorithms [12].

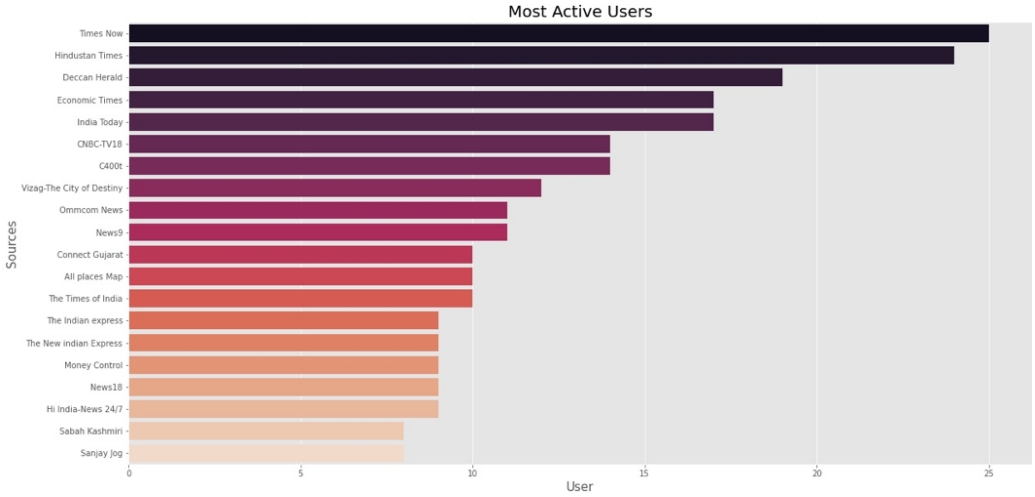

Figure 8. This specific statistic displays the most active accounts on Twitter after their tweets have been cleaned.

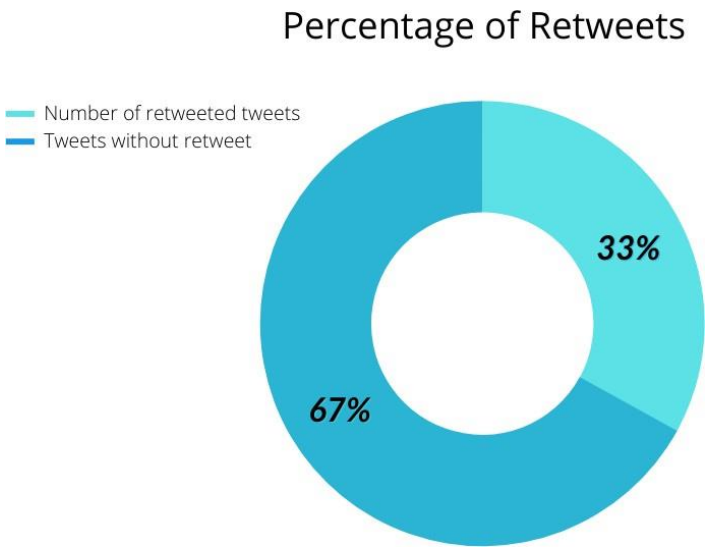

Figure 9. After the cleaning of dataset, we performed a differentiating tedious task over the tweets which gave us an overview that 33% of tweets in this donut graph had retweets from other users.

V. CLUSTERING

Iterative methods are often used in the clustering process to group instances together based on shared features. Using these categories, we may examine the data, detect anomalous behavior, and then anticipate outcomes. When it comes to the visualization of clusters, the models of clustering may also assist you to discover connections that you might not be able to infer from simple browsing or just watching. Since some of these causes, clustering is often employed in machine learning projects in the early phases of the project to explore the data and find interesting results.

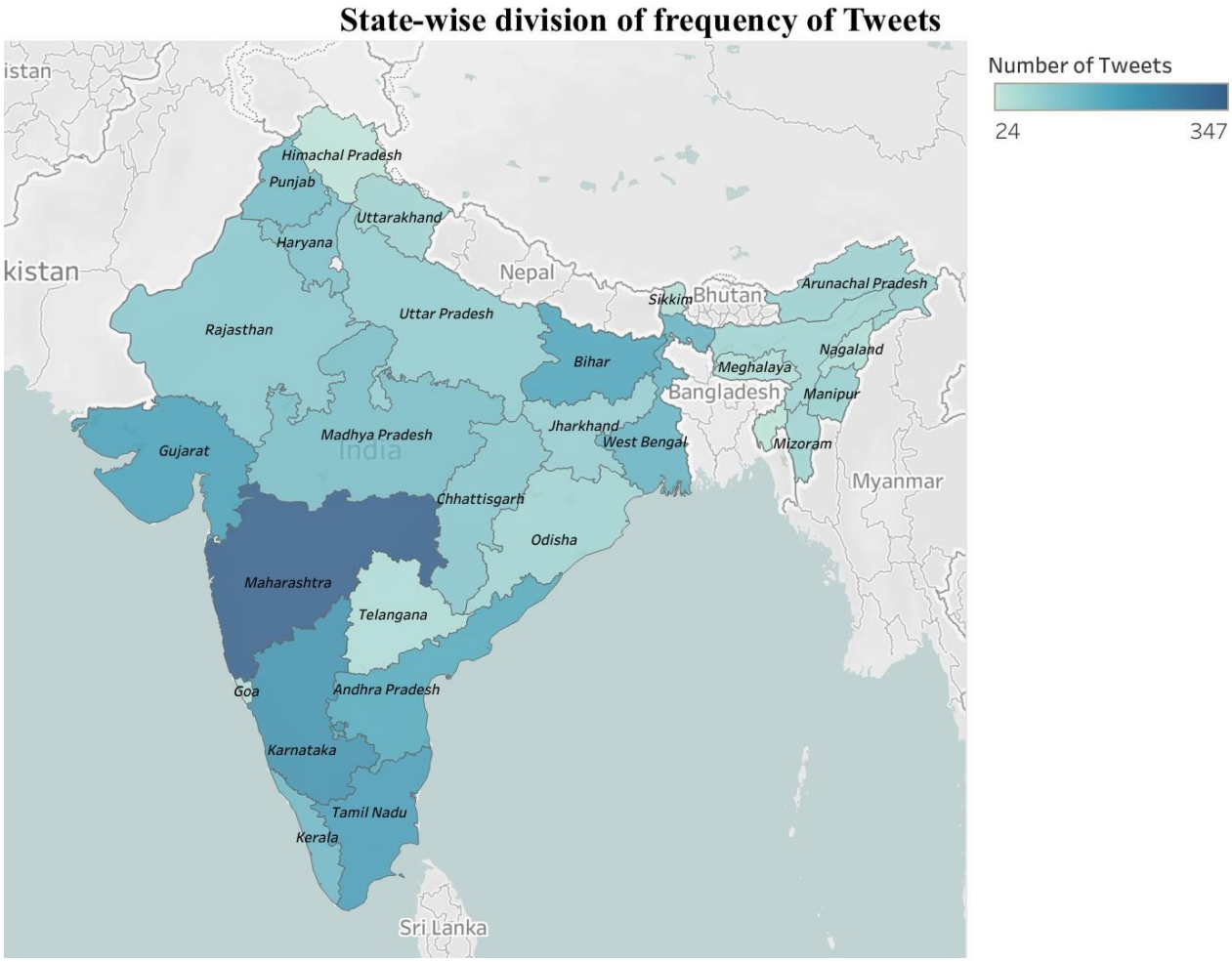

Figure 10. This is a location-based tweet frequency map that shows how often people tweeted from each state in India, except Jammu & Kashmir (Negligible Tweets from that area). Citizens in Maharashtra being the most active in relation to their activity over Twitter.

Having done all the preparation of tweets, we are now left with 9525 tweets on which we intend to apply to a cluster, Figure 11. In this stage of development, the data did say that we have prepared, we apply k-means clustering over it [18]. This method is aimed at partitioning n observations into k clusters, each of which has the closest mean value. Though I have to admit, working in n-dimensional areas is critical. If the effort fails, a part of the sample is shifted until each sample is closer to the cluster center.

Elbow Method and Silhouette Method:

To identify K clusters [18], we can see the 3 clusters established using the silhouette technique are shown. Based on these findings, we decreased the dimensions, and then assigned their domains as well as colors to distinguish them based on the cluster values, and setting K centroids to distinct values. The overall variance between each group decreases as we add additional clusters. These findings, therefore, do not demonstrate the ideal number of K clusters to seek. To achieve the correct K number of centroids for our prediction analysis, we must decrease the within-cluster sum of squares. To do so, we first utilized the Elbow technique to determine the ideal number of clusters [19], [20]. The graph and the illustration below are provided so that you can see the effect of varying the number of clusters using the Elbow technique, Figure 12. When taking the total of squares inside groups, one would expect to see 2 at the end of the elbow.

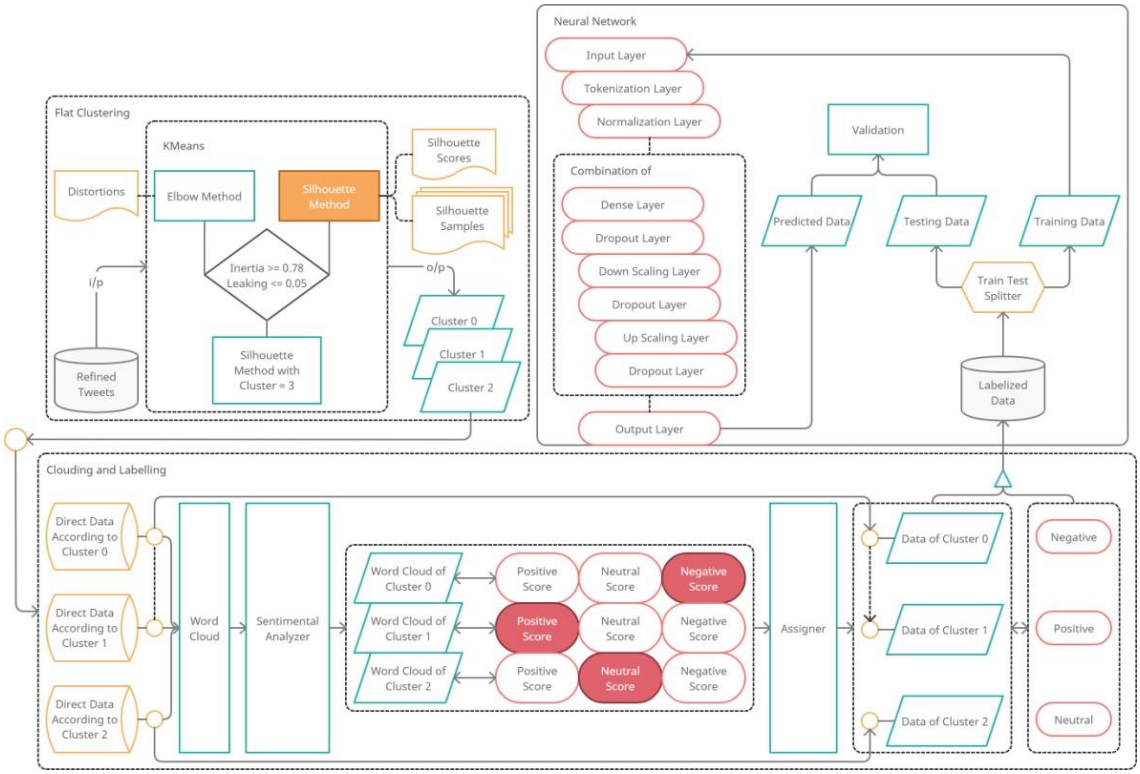

Figure 11. The diagram above depicts the transmission of data in flat clustering over three clusters which were predicted using the Silhouette Method. Following that, we create a word cloud and labelize various words into sentiments. All of this is then predicted and validated using a neural network.

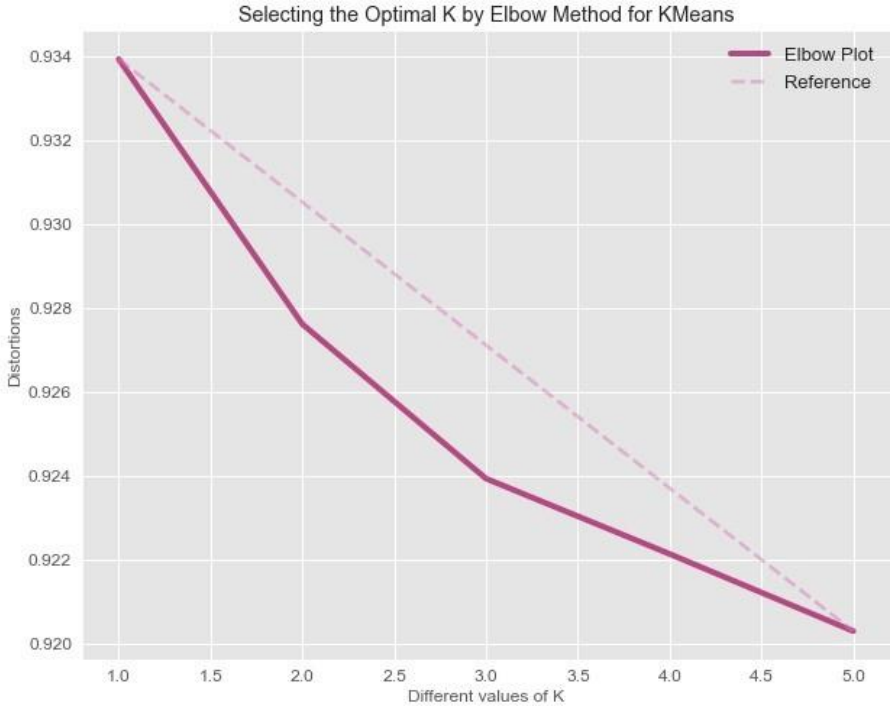

Figure 12. The figure above illustrates the Elbow Method, which determines the optimum number of clusters using the elbows. The fact that this presents us with an elbow at 2 leaves us with some ambiguity

When you look at K equals 2, it seems to be an elbow. Adding the number of clusters to the within-group mean square would therefore result in a much worse answer [21], [22]. In this example, the best number of cluster centers is two. However, the narrative is open to interpretation, and although we aren't sure due to having the other point at 3, this further weakens our case. For that, we used the second strategy, the 'Silhouette technique,' which ensured that the correct amount of k clusters were created [23]. The figure below illustrates the Silhouette method's optimum number of clusters, Figure 13.

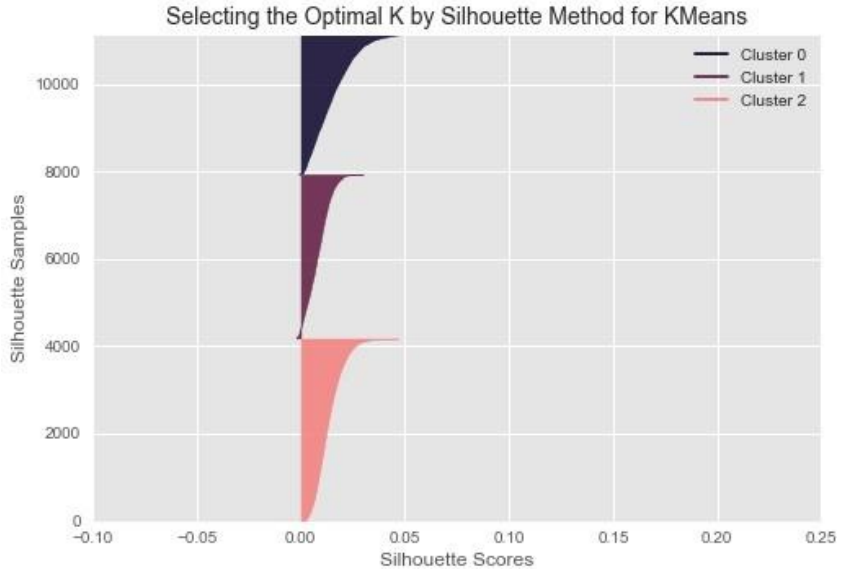

Figure 13. We choose Silhouette, which presents us with a close approximation of the number of clusters: three. In practice, since there is no negative cohesiveness, this is optimum cluster number.

To understand and validate consistency among data clusters, one may use the silhouette technique. It is determined by the closeness of an entity from its own (cohesion) cluster against the distance of that entity to all other (separation) clusters. A high score indicates that the item is quite well linked with its cluster [24], [25], Whereas a low score indicates that it is poorly matched to adjacent clusters. Setting up the clusters is proper if most items have a high value. However, if there are a significant number of points with a negative or negligible value, there may be an overabundance or deficit of clusters in the clustering arrangement. Observe in the following diagram that cluster number 2 has a negative value. When we examine the opposite side, we see that each point is bright and consistent in groups, and without disruption. Our prediction methodology relies on 3 clusters, which, thus, may be inferred to be the most optimum, Figure 14.

Simple counts provide a challenge because of how common some words like "the" are, and these high counts in the encoded vectors will be a small fraction of the total. There are several different ways to go about finding word frequencies. One of the most common is TF-IDF (Term Frequency–Inverse Document) [26]. A TF-IDF is assigned to each word to generate the score. Frequency indicates the appearance of a certain word in a given text. Words that often appear in several files TF-IDF uses word frequency scores to highlight more interesting terms, for example, how often a word occurs in a particular file, but not across other files. Tokenization, vocabulary learning, and document frequency weighting are all part of the Tfidf Vectorizer's capabilities. A Tfidf Transformer is often used for monitoring the number of times text is reversed and the text encoding process is started. Normalized scores are typically in the range of 0 to 1, which allows for ready usage of encoded document vectors.

K-means clusters the documents, and TF-IDF always measures a non-negative value, thus every document in a cluster will include its centroid, which is equal to the mean of all documents in the cluster. So, in other words, the keywords with the greatest impact on the centroid are those that have the most impact throughout all of the papers in that cluster. Each word is included, but a lot of unimportant information is ignored [26]. In other words, words most important to the document vector have the greatest TF-IDF values Equation (1), while those most important to the cluster as a whole have the highest centroid values.

$$tf_{t,d} = \frac{count_{t,d}}{totalcount_d} \quad [1]$$

The total count (the entire number of all words in the text) is known as count (t, d). When IDF evaluates the degree to which a word is informative in a text for model training, it's taking into consideration many different dimensions. One may calculate it as Equation (2).

$$idf = \frac{N}{Df_t} \quad [2]$$

The N-by-Dft matrix represents the total number of documents in the corpus that include the word t. When a word often appears in multiple texts, IDF assesses the weight of the term. An example of this is that stop words do have low IDF scores. As shown before, TF-IDF may be described as Equation (3).

$$tf - idf = tf_{t,d} \times \log(idf) \quad [3]$$

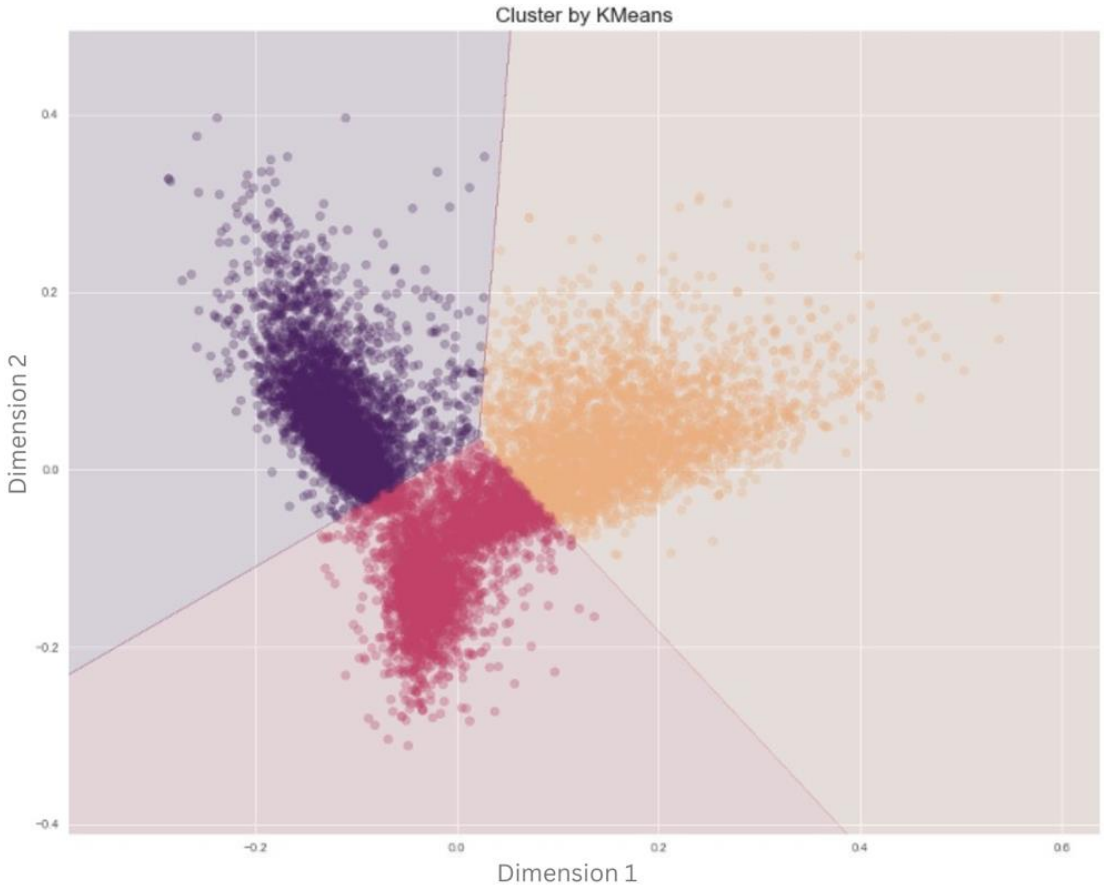

Figure 14. Here we can see the 3 clusters established using the silhouette technique are shown. Based on these findings, we decreased the dimensions, and then assigned their domains as well as colours to distinguish them based on the cluster values

When dealing with large datasets, it's always difficult to display the data, in our instance, tweets, and therefore very difficult to glean information from our tweets. As a result, it becomes our primary objective to decrease this greater dimensionality, which may be accomplished using methods such as PCA or TruncatedSVD [27], [28]. Either of the functions contributes to the reduction of the number of dimensions, which results in the elimination of outliers and the efficiency of calculation. It is usually suggested utilizing PCA or TruncatedSVD to decrease the number of dimensions to a manageable level when the number of features is extremely large [29]. This really does contribute significantly to noise reduction and acceleration in the calculation of pairwise distances between samples.

VI. WORD CLOUD AND LABELING

A graphical depiction of tweets was produced by creating word clouds that help to show words as they arrive in tweets. Word clouds use the frequency of words to emphasize them. Throughout this study, the frequently performed word clouds in tweets linked to COVID-19 and its vaccination gave greater insight regarding the tweets about vaccine and COVID-19 over Twitter [30]. Based on the Figure 15, which is the preclustering word cloud with the information of all the words obtained from the preprocessed Twitter tweets, other very frequently present terms were associated with Vaccine, Vaccination, Age, and Group. These words correspond to the citizens' desire for vaccinations and concern about mandatory vaccination rules based on age group and the requirements being associated with COVID-19. In addition, the term people and death illustrate the distribution of the virus to people, along with their thoughts and fears, while the word 'Price' depicts the conditions that affect people's finances due to the continuing spread of the infection. There are eighteen, forty-five, state, and forty-four terms that are included in our word cloud to represent public perceptions of our research [31], [32], Figure 16.

We have two goals for this project: We want to analyze the impact of vaccines, and we want to learn more about people's emotions in general by looking at popular tweets which mention people ages 18-45 who need immunizations. Now we have three differentiating and distanced word clusters to look forward to. However, we did not know which cluster had positive, negative, and neutral terms. The following procedure shows how to assign labels to these clusters: SentimentIntensityAnalyzer, which is contained in the NLP package NLTK, is utilized [12].

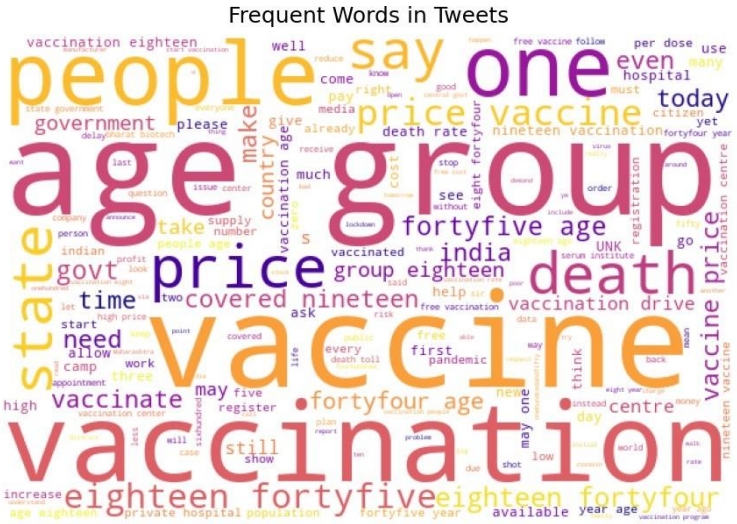

Figure 15. Here we see a word cloud of tweets that have not been clustered, which illustrates all the feelings and emotions of Twitter users.

This returns a dictionary having a collection of various scores there were 3 scores: negative, neutral, and positive. All of which added up to 1 and the sum couldn't be negative. To give an example, let's use the text "Wow, NLTK is extremely strong" with the algorithm pass. The algorithm produces 0.0, 0.295, 0.705, and 0.8012. The resultant of all three first measurements were negative, neutral, and positive and the last one was the compound in which the readings range from -1 to 1. This is used to identify and classify clusters based on which variable has the greatest level [25].

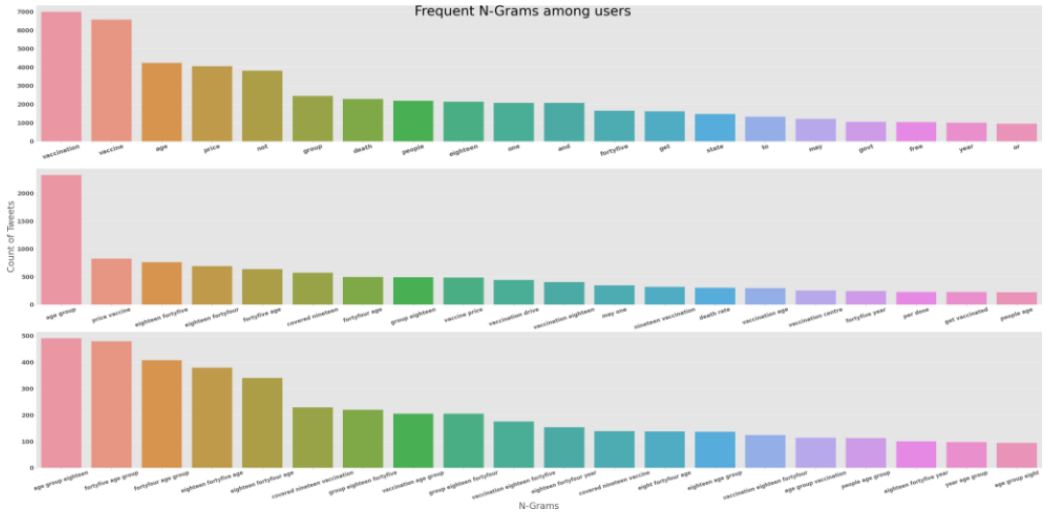

Figure 16. Unigram, Bigram, Trigram, and so on make up a frequent n-gram. We can observe a nice visual validation of the Unigram that pertains to vaccinations, age, pricing, and more, providing us confidence that the dataset is clean. In both cases, bigrams such as age group, price vaccine, and so on are listed as the concerns. Additionally, we see trigrams like "age group eighteen", "age group forty-five," and so on all reflecting the main problem as well.

The results show conflicting feelings when looking at Cluster 1 in the Word-cloud for all the Negative terms cluster [33], [34], [35]. Anger and confusion are very well expressed when it comes to this topic. Illegal is used to indicate that individuals in India have been receiving vaccine doses in different methods which aren't provided by the government and the citizens have been discussing it. We also look at the other words like disaster, apathy, delay, chaotic, suspend, terrible, scam, liar, etc. words used by people expressing their emotions towards this issue, Figure 17.

Cluster 2 for Positive words has been speaking more about Increase & Guarantee, which are national ideology-related to the government's ability to ensure everyone gets a vaccine [34]. We can note that citizens have been talking about hope and expectations arising from an increase in vaccination. We can observe the terms "freedom," "awesome," "effective," "happily," "special," "pretty," and other words related to happiness appear as people are overjoyed with this choice, which allows them to see hope and to express their enthusiasm. We get a glimpse of how this choice has made it simpler for the dozes to be obtained via the term "easy", Figure 17.

Figure 17. The image above is composed of 3 components, meaning essentially word cloud creation after being analyzed & clustered for the various emotions about the words in tweets. A] The leftmost being cluster over negative words, B] The one in center being cluster over positive words, C] Rightmost word cloud representing neutral words.

Neural words tend to be hard to portray emotionally since they rely on neither positive nor negative [33], [35]. Despite these being matters of personal or public safety, we observe that there are discussions through tweets linked to senior citizens, dogs, institutions, booths, etc. Figure 17.

VII. NEURAL NETWORK

A deep learning neural network is composed of multiple hidden nodes and therefore is inspired by the brain's neural networks, which are used to provide such a dependable output. This is useful for improving the precision of the Tweet user's feelings. This neural network is built using the TensorFlow framework [36]. The positive and negative keywords are first processed, and then the data from these processing stages is stored for comparisons. Tweets that have been analyzed before their publication are inspected for terms that are associated with either positive or negative sentiment, and then those tweets are assigned either to positive, negative, or neutral. Each tweet with a good, negative, or neutral message will have a corresponding score of >=1, <=-1, or =0 accordingly. So, using the Neural Network we had lots of variations in our Models. Starting from 1st to getting an accuracy of 97% in the 5th and our final Model of Neural Network.

1st Model: At the beginning, we started by three dense layers connected t 2^14 hidden layers. The categorical cross-entropy loss function and SoftMax activation function are used in this research for multi-class and single-label predictions. We used tokenization, sequential embedding, and dense layer network building blocks to construct our network. This model is applied to the data that is remaining after training and is also validated on the validation set. We observe that the training loss decreases slowly while the validation loss continues to increase from epoch 5, Table 1. As both of them must increase together to get a good result. We conclude, that we had an overfitting problem in this model thus we went forward with another model.

2nd Model: To deal with these issues, we included dropout layers into the network model to help alleviate overfitting. Now we apply the dropout layer to the model to help with accuracy. LSTM is a kind of recurrent neural network, and like with RNN [37], it is a far more powerful choice when you require the network to remember information for a longer duration. We observe a slight variation in our accuracy but it's still not as good as we required because this well is having an overfitting problem, Table 1.

3rd Model: As a result, we go to the next model where we utilize various sampling schemes to improve the model's accuracy. After 10 epochs, the model begins to overfit and upscaling and downscaling begin again, Table 1. There was a rise in the rate of loss before models, though.

4th Model: We now use a bidirectional LSTM model to improve the model's accuracy [37]. Traditional LSTM help enhance model performance on sequence classification tasks, and these bidirectional LSTM are an extension of such models. Here we had better results when it came to training accuracy [38], [39], but when it came to real-world performance our F1 scores were poor and the model was unable to accurately predict the emotions of the tweets, Table 1.

5th Model: We ended up using dropouts, normalization, and several sampling approaches to enhance the accuracy of our model [36], [39]. This all led to excellent F1 ratings for each label, which also allowed us to pinpoint our issue statement, Figures 18 and 19.

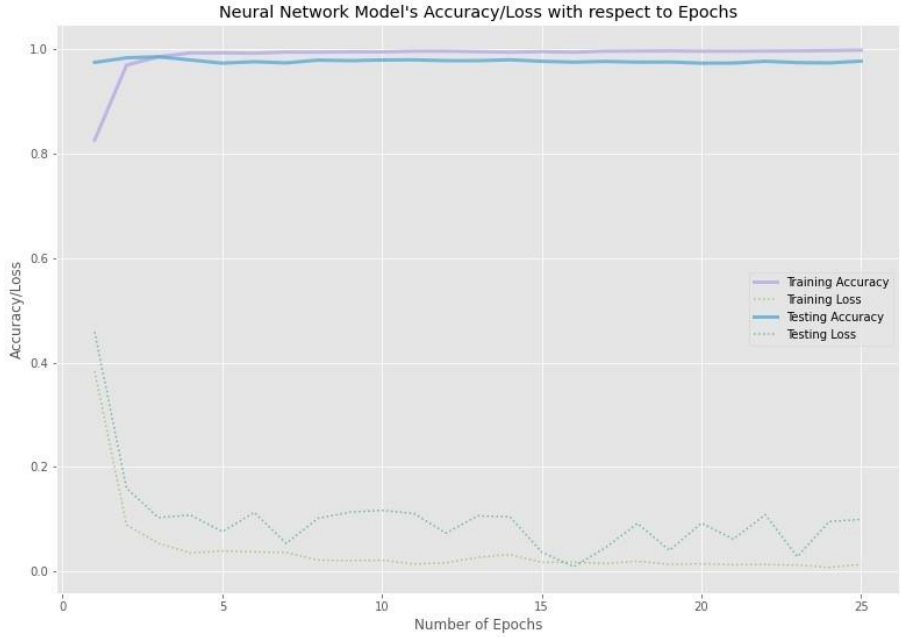

Figure 18. In this figure, we took a model and fed the data to the Neural Network Architecture where it does forward and backward propagation, with respect to the epoch. We can see how the model reduces its losses and gains accuracy over the testing and training set.

Table 1. As can be seen from this table, there is a distinction between the matrices. We worked with a variety of models, but for the comparison, we chose five of the most optimal models. They are compared on their set levels over the matrix [Average Scores, F1 scores, Precision & Recall] in their respective classes.

Neural Network		Model 1	Model 2	Model 3	Model 4	Model 5
Average Scores	Training	0.99	0.98	0.97	0.95	0.95
	Testing	0.97	0.95	0.93	0.93	0.91
F1 scores	Class 0	0.98	0.96	0.97	0.95	0.91
	Class 1	0.96	0.93	0.95	0.93	0.88
	Class 2	0.96	0.95	0.95	0.92	0.89
Precision	Class 0	0.99	0.95	0.95	0.92	0.89
	Class 1	0.95	0.9	0.96	0.9	0.92
	Class 2	0.97	0.89	0.92	0.93	0.91
Recall	Class 0	0.97	0.9	0.89	0.9	0.91
	Class 1	0.95	0.91	0.88	0.91	0.92
	Class 2	0.97	0.89	0.93	0.9	0.92

VIII. RESULT

The project's main aim was to analyze Vaccination tweets during the Covid-19 epidemic for emotion. We began taking a dataset of approximately 50,000 tweets from Twitter, and we worked our way through it to finally get a dataset of about 9525 tweets, Figure 1. In the process, we had found that only about 20% of the dataset were duplicates, and our tweets were especially relevant to the issue we had set out to study. A loss of data may be a concern, but it is much more critical to ensure that the cleanest possible clusters are created. Our final model, the Neural Network, trained with 99% accuracy and testing resulted in an overall rating of 97%Figures 18 and 19. A high level of accuracy concerning both class 1 and class 2 at 96% for both and a high level of precision for class 0 at 98%. The precision ranged from 95% and topped to 99% for the classes.

As a result, this became feasible, because we followed steps where we took the tweets, performed filtering over them during the data gathering process, where we filtered over our 724 keywords in hashtags out of a dataset aggregately having 2657 hashtags. As a result, we've received 30% of the total number of tweets as 15,174, i.e., 30% of the dataset. Removing URLs, Hashtags, utilizing Regular expressions, Tokenization and Stop words using NLTK [12]. 100% of the words were converted from their derived forms to their base forms because of the Lemmatization. Everything in this presentation was obtained in three different versions. As Version 3 was the most optimized, we calculated that our Version 1 for 25 tweets took about 30 seconds. While our most optimized version was 600% faster. This task, on average, took 0.2 seconds to do, Figure 3.

Our support count used to decrease due to the tweets by users, just being copied and appended with some extra words. Without removing this, we would have ended up with flawed clusters and irregularities, Figure 6. Ideating and creating an algorithm and having versions of it, we removed these distortions not just with accuracy but also with the least time taken. Our first comparison took 2 mins 20 secs to clean, while we beat the time scores with our accuracy being 100% and time is taken comparatively at 3*10^4 times faster, only taking 0.5 secs for 500 tweets, Figure 7.

We found that using k-means clustering, our leaking factor was minimum, and validated the number of clusters using the Silhouette Method. It was observed that the silhouette method performed far better than the Elbow Method, which helped in forming profoundly distinguishable clusters, Figures 13 and 14. Thus, then all of this made it suitable for the categorization of the 9525, cleaned tweets which show that 33% are negatively classified, and 33% are positive, while the rest being neutral, Figure 17.

IX. CONCLUSION

A systematic study of Covid-19 emotions stated in tweets by users is reported in this paper. Given both the worldwide escalation of the pandemic and the changing perceptions of the virus's consequences, such studies like we provide are becoming more important for researchers within the medical field in matters ranging from health issues to public awareness. This example illustrates the ability to summarize beliefs regarding experimentally validated disease preventive strategies via mining Twitter data.

We used several preprocessing methods and feature extraction techniques, beginning with the 49,345-row dataset, which got cleaned to 9525 Tweets, Figure 7, lastly used to train the Neural Network model. Immediate estimates of what people are saying and experiencing throughout the viral devastation may be produced with the assistance of abstract visualizations, such as the clustering methods shown. With using K-Means Clustering we could say that it aids classification well, given the tweets must be clean. We observe a success rate of 99% after a successful training and testing process. In researching F1 scoring, the literature has revealed scores of almost 98 percent, Figure 18. Using the approach, we conclude that with our preprocessing method, algorithm, and Neural network model we developed, we can be certain that our findings will continue to be optimal as the number of tweets about vaccines continues to grow day by day.

X. FUTURE SCOPE

Our pipeline is designed to serve as a helpful and beneficial tool for healthcare professionals and public officials by presenting useful information about global health issues. In the analysis, one may determine which groups or subgroups have fallen short of being reached by current programs and make use of predictive modeling to help health organizations plan and optimize outreach initiatives.

Additionally, in the current research, the time of vaccination news and updates is limited to just a certain duration. Vaccine development stages, projection of vaccine availability, vaccination approvals, and early vaccine doses have taken the bulk of Twitter activity during this time period. Future studies may incorporate vaccine-related tweets as individuals in the age range over 18 are currently getting vaccinations.

REFERENCES

[1] Yang, J., Chen, X., Deng, X., Chen, Z., Gong, H., Yan, H., Wu, Q., Shi, H., Lai, S., Ajelli, M., Viboud, C., & Yu, P. H. (2020). Disease burden and clinical severity of the first pandemic wave of COVID-19 in Wuhan, China. Nature Communications, 1. https://doi.org/10.1038/s41467-020-19238-2.

[2] Li, L., Zhang, Q., Wang, X., Zhang, J., Wang, T., Gao, T.-L., Duan, W., Tsoi, K. K., & Wang, F.-Y. (2020). Characterizing the Propagation of Situational Information in social media During COVID-19 Epidemic: A Case Study on Weibo. IEEE Transactions on Computational Social Systems, 2, 556–562. https://doi.org/10.1109/tcss.2020.2980007

[3] Chamola, V., Hassija, V., Gupta, V., & Guizani, M. (2020). A Comprehensive Review of the COVID-19 Pandemic and the Role of IoT, Drones, AI, Blockchain, and 5G in Managing its Impact. IEEE Access, 90225–90265. https://doi.org/10.1109/access.2020.2992341.

[4] Long, S. W., Olsen, R. J., Christensen, P. A., Bernard, D. W., Davis, et al. (2020). Molecular Architecture of Early Dissemination and Massive Second Wave of the SARS-CoV-2 Virus in a Major Metropolitan Area. MBio, 6. https://doi.org/10.1128/mbio.02707-20.

[5] Salzberger, B., Glück, T., & Ehrenstein, B. (2020). Successful containment of COVID-19: the WHO-Report on the COVID-19 outbreak in China. Infection, 2, 151–153. https://doi.org/10.1007/s15010-020-01409-4.

[6] Vahidy, F. S., Drews, A. L., Masud, F. N., Schwartz, R. L., Askary, B. "Billy," Boom, M. L., & Phillips, R. A. (2020). Characteristics and Outcomes of COVID-19 Patients During Initial Peak and Resurgence in the Houston Metropolitan Area. JAMA, 10, 998. https://doi.org/10.1001/jama.2020.15301.

[7] Zhu, N., Zhang, D., Wang, W., Li, X., Yang, B., Song, J., et al. (2020). A Novel Coronavirus from Patients with Pneumonia in China, 2019. New England Journal of Medicine, 8, 727–733. https://doi.org/10.1056/nejmoa2001017.

[8] Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New Avenues in Opinion Mining and Sentiment Analysis. IEEE Intelligent Systems, 2, 15–21. https://doi.org/10.1109/mis.2013.30.

[9] Shen, K.-L., Yang, Y.-H., Jiang, R.-M., Wang, T.-Y., Zhao, D.-C., et al. (2020). Updated diagnosis, treatment and prevention of COVID-19 in children: experts' consensus statement (condensed version of the second edition). World Journal of Pediatrics, 3, 232–239. https://doi.org/10.1007/s12519-020-00362-4.

[10] Cotfas, L.-A., Delcea, C., Roxin, I., Ioanas, C., Gherai, D. S., & Tajariol, F. (2021). The Longest Month: Analyzing COVID-19 Vaccination Opinions Dynamics from Tweets in the Month Following the First Vaccine Announcement. IEEE Access, 33203–33223. https://doi.org/10.1109/access.2021.3059821.

[11] Fan, G., Yang, Z., Lin, Q., Zhao, S., Yang, L., & He, D. (2020). Decreased Case Fatality Rate of COVID-19 in the Second Wave: A study in 53 countries or regions. Transboundary and Emerging Diseases, 2, 213–215. https://doi.org/10.1111/tbed.13819.

[12] Jongeling, R., Datta, S., & Serebrenik, A. (2015). Choosing your weapons: On sentiment analysis tools for software engineering research. In 2015 IEEE International Conference on Software Maintenance and Evolution (ICSME) (pp. 531-535).

[13] Liu B. (2011) Opinion Mining and Sentiment Analysis. In: Web Data Mining. Data-Centric Systems and Applications. Springer, Berlin, Heidelberg.

[14] Liu B., Zhang L. (2012) A Survey of Opinion Mining and Sentiment Analysis. In: Aggarwal C., Zhai C. (eds) Mining Text Data. Springer, Boston, MA.

[15] Goel, A., Gautam, J., & Kumar, S. (2016). Real time sentiment analysis of tweets using Naive Bayes. In 2016 2nd International Conference on Next Generation Computing Technologies (NGCT) (pp. 257-261).
[16] Nelli F. (2018) Textual Data Analysis with NLTK. In: Python Data Analytics. Apress, Berkeley, CA.
[17] Yogish D., Manjunath T.N., Hegadi R.S. (2019) Review on Natural Language Processing Trends and Techniques Using NLTK. In: Santosh K., Hegadi R. (eds) Recent Trends in Image Processing and Pattern Recognition. RTIP2R (2018). Communications in Computer and Information Science, vol 1037. Springer, Singapore.
[18] Likas, A., Vlassis, N., & J. Verbeek, J. (2003). The global k-means clustering algorithm. Pattern Recognition, 2, 451–461. https://doi.org/10.1016/s0031-3203(02)00060-2.
[19] Purnima Bholowalia, & Arvind Kumar (2014). Article: EBK-Means: A Clustering Technique based on Elbow Method and K-Means in WSN. International Journal of Computer Applications, 105(9), 17-24.
[20] Yuan, Chunhui & Yang, Haitao. (2019). Research on K-Value Selection Method of K-Means Clustering Algorithm. J. 2. 226-235. 10.3390/j2020016.
[21] Aranganayagi, S., & Thangavel, K. (2007). Clustering Categorical Data Using Silhouette Coefficient as a Relocating Measure. In International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007) (pp. 13-17).
[22] Thinsungnoen, Tippaya & Kaoungku, Nuntawut & Durongdumronchai, Pongsakorn & Kerdprasop, Kittisak & Kerdprasop, Nittaya. (2015). The Clustering Validity with Silhouette and Sum of Squared Errors. 44-51. 10.12792/iciae2015.012.
[23] Kodinariya, Trupti & Makwana, Prashant. (2013). Review on Determining of Cluster in K-means Clustering. International Journal of Advance Research in Computer Science and Management Studies. 1. 90-95.
[24] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. Pattern Recognition Letters, 8, 651–666. https://doi.org/10.1016/j.patrec.2009.09.011.
[25] Wagstaff, Kiri & Cardie, Claire & Rogers, Seth & Schrödl, Stefan. (2001). Constrained K-means Clustering with Background Knowledge. Proceedings of 18th International Conference on Machine Learning. 577-584.
[26] Zhang, W., Yoshida, T., & Tang, X. (2011). A comparative study of TF*IDF, LSI and multi-words for text classification. Expert Systems with Applications, 3, 2758–2765. https://doi.org/10.1016/j.eswa.2010.08.066.
[27] Sehgal, S., Singh, H., Agarwal, M., Bhasker, V., & Shantanu (2014). Data analysis using principal component analysis. In 2014 International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom) (pp. 45-48).
[28] Martinez, A. M., & Kak, A. C. (2001). PCA versus LDA. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2, 228–233. https://doi.org/10.1109/34.908974.
[29] Yan, S., Xu, D., Zhang, B., Zhang, H., Yang, Q., & Lin, S. (2007). Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence,1, 40–51, https://doi.org/10.1109/tpami.2007.250598.
[30] Heimerl, F., Lohmann, S., Lange, S., & Ertl, T. (2014). Word Cloud Explorer: Text Analytics Based on Word Clouds. In 2014 47th Hawaii International Conference on System Sciences (pp. 1833-1842).
[31] Pak, Alexander & Paroubek, Patrick. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. Proceedings of LREC. 10.
[32] Saif H., He Y., Alani H. (2012). Semantic Sentiment Analysis of Twitter. In: Cudré-Mauroux P. et al. (eds) The Semantic Web – ISWC 2012. ISWC 2012. Lecture Notes in Computer Science, vol 7649. Springer, Berlin, Heidelberg.
[33] Jelodar, H., Wang, Y., Orji, R., & Huang, S. (2020). Deep Sentiment Classification and Topic Discovery on Novel Coronavirus or COVID-19 Online Discussions: NLP Using LSTM Recurrent Neural Network Approach. IEEE Journal of Biomedical and Health Informatics, 24(10), 2733-2742.
[34] Wang, X., Ma, X., & Grimson, E. (2007). Unsupervised Activity Perception by Hierarchical Bayesian Models. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8).
[35] Blei, David & Ng, Andrew & Jordan, Michael. (2001). Latent Dirichlet Allocation. The Journal of Machine Learning Research. 3. 601-608.
[36] Erik Wiener, Jan O. Pedersen, & Andreas S. Weigend. (1995). A Neural Network Approach to Topic Spotting.
[37] Zaremba, Wojciech & Sutskever, Ilya & Vinyals, Oriol. (2014). Recurrent Neural Network Regularization.
[38] Tsai, J.T., Chou, J.H., Liu, T.K (2006). Tuning the structure and parameters of a neural network by using hybrid Taguchi-genetic algorithm. IEEE Transactions on Neural Networks, 17(1), 69-80.
[39] Leung, F., Lam, H., Ling, S., & Tam, P. (2003). Tuning of the structure and parameters of a neural network using an improved genetic algorithm. IEEE Transactions on Neural Networks, 14(1), 79-88.