Scientific Research Publishing

Open Journal of Statistics, 2021, 11, 620-632
https://www.scirp.org/journal/ojs
ISSN Online: 2161-7198
ISSN Print: 2161-718X

Machine Learning Approaches for Classifying the Distribution of Covid-19 Sentiments

M. Kuyo[1], S. Mwalili[2], E. Okang'o[3]

[1]Jomo Kenyatta Unversity of Agriculture and Technology, Nairobi, Kenya
[2]Department of Statistics and Actuarial Sciences, JKUAT, Nairobi, Kenya
[3]Department of Mathematics and Actuarial Sciences, Murang'a University of Technology, Murang'a, Kenya
Email: kaitikeikuyo@gmail.com, samuel.mwalili@gmail.com, kangphas@gmail.com

How to cite this paper: Kuyo, M., Mwalili, S. and Okang'o, E. (2021) Machine Learning Approaches for Classifying the Distribution of Covid-19 Sentiments. *Open Journal of Statistics*, **11**, 620-632.
https://doi.org/10.4236/ojs.2021.115037

Received: August 13, 2021
Accepted: September 27, 2021
Published: September 30, 2021

Abstract

Previously, rapid disease detection and prevention was difficult. This is because disease modeling and prediction was dependent on a manually obtained dataset that includes use of survey. With the increased use of social media platforms like Twitter, Facebook, Instagram, etc., data mining and sentiment analysis can help avoid diseases. Sentiment analysis is a powerful tool for analyzing people's perceptions, emotions, value assessments, attitudes, and feelings as expressed in texts. The purpose of this research is to use machine learning techniques to classify and predict the spatial distribution of positive and negative sentiments of Covid-19 pandemic. This study research has employed machine learning to classify spatial distribution of Covid-19 twitter sentiments as positive or negative. The data for this study were geo-tagged tweets concerning COVID-19 which were live streamed using streamR package. The key terms used for streaming the data were: Corona, Covid-19, sanitizer, virus, lockdown, quarantine, and social distance. The classification used Naive Bayes algorithms with ngram approaches. N-Gram model is a probabilistic language model used to predict next item in a sequence in the form (n − 1) order Markov. It relies on the Markov assumption—the probability of a word depends only on the previous word without looking too far into the past. The steps followed in this research include: cleaning and preprocessing the data, text tokenization using n-gram *i.e.* 1-gram, 2-gram, and 3-gram, tweets were converted or weighted into a matrix of numeric vectors using Term Frequency Inverse-Document. Also, data were divided 80:20 between train and test data. A confusion matrix was utilized to evaluate the classification accuracy, precision, and recall performance of the various algorithms tested. Prediction was done using the best performing Naive Bayes algorithm. The results of this research showed that under Multinomial Naive Bayes, unigram accuracy was 92.02%, bigram accuracy was 97.37%, and trigram accu-

racy was 94.40%. Unigram had 89.34% accuracy, bigram had 96.80%, and trigram had 94.90% accuracy using Bernoulli Naive Bayes. Unigram accuracy was 90.43%, bigram accuracy was 95.67%, and trigram accuracy was 92.89% using Gaussian Naive Bayes. Bigram tokenization outperformed unigram and trigram tokenization. Bigram Multinomial Naive Bayes was used to predict test data since it was the most accurate in classifying train data. Prediction accuracy was 84.92%, precision 85.50%, recall 81.02%, and F1 measure 83.20%. TF-IDF was employed to increase prediction accuracy, obtaining 87.06%. These were then plotted on a globe map. The study indicates that machine learning can identify patterns and emotions in public tweets, which may then be used to steer targeted intervention programs aimed at limiting disease spread.

Keywords

Machine Learning, Sentiment Analysis, Natural Language Processing, Covid-19, Naive Bayes, N-Gram

1. Introduction

In the past years, disease prediction was done based on a conventional dataset and modelled using traditional approaches. This is because disease modeling and prediction was dependent on a manually obtained dataset that includes use of survey. Researchers have had a great need to establish effective analytical approaches to comprehend the flow of information and improve human perceptions under epidemic situations since the rapid spread of Covid-19 (Coronavirus) infection [1] [2]. It's crucial to remember that epidemics have afflicted the planet for millennia, and the consequences of those pandemics have had a significant impact on the world and resulted in countless deaths [3]. As a result, in order to make reliable predictions, it is necessary to understand the disease's natural development. While several initiatives are utilized to collect and evaluate data during pandemics, most studies are now focusing on text mining using Twitter data [4]. The increased usage of textual analytics, natural language processing (NLP), and its applications [2] are driving this trend [4]. In case of a pandemic, conventional data collection methods are hampered, and time barred. As such other data sources like social media data hold valuable actionable information that can help inform intervention strategies [3]. Natural Language Processing (NLP) application in sentiment analysis and text mining from social media offers a powerful tool that can be used for data collection during pandemics. Sentiment analysis is a powerful tool for analyzing many issues related to human interaction with the computer [5]. In recent studies, this concept has been extended to sociology, advertising, marketing, and healthcare [6].

Sentiment analysis is the process of mining ideas by analyzing people's feelings, emotions, value evaluations, attitudes, and their sentiments [7]. Sentiment

analysis study was previously nonexistent because there was no opinion document in digital form prior to the early 2000s [4]. Now that the internet and social media have exploded in popularity over the past fifteen years, we have a never-ending flow of information about ideas that exist in digital form [4]. [8] gathered raw data on COVID-19 outbreaks and used Latent Dirichlet Allocation (LDA) data in the datasets' document term matrix. The LDA's technique found that the novel's COVID-19 outbreak was characterized by negative emotions such as fear and good emotions such as trust. [9] created a list of COVID-19-related hashtags and used them to search for relevant tweets for two weeks starting on January 14, 2020. The keywords linked with the tweets are recognized and saved as text using an API. The emotional valence of the tweet, *i.e.* (good, bad, and neutral), as well as prominent emotions (anger, fear, happiness, disgust, sadness, or surprise) were discovered after undertaking an emotional evaluation of infection preventive techniques, immunizations, and racial prejudice.

Sentiment analysis offers several approaches for sentiment classification: machine learning approach, a lexicon-based approach, and a hybrid approach. Machine Learning (ML) uses standard ML techniques and language features. ML methods are split into supervised and un-supervised learning methods [10]. The algorithm will learn from a data that is labeled in a supervised learning method while in an unsupervised learning model, the algorithm is provided with a non-labeled data that the algorithm attempts to make sense by extracting the features and patterns themselves [10]. Lexicon-based strategy focuses on an emotional vocabulary, a collection of renowned and established words. The Hybrid Approach (HA) blends machine learning and a lexical approach. HA is omnipresent with lexicons of feeling and is particularly useful in most ways.

Corona virus, a unique disease that has unexpectedly escalated into a pandemic called COVID-19. On December 31, 2019, WHO declared it on a Wuhan province in China, and it swiftly reached practically every country. The coronavirus-2 has a detrimental effect on the respiratory system, which results in a severe form of the common cold [11]. Coughing and fever are among the common signs of the sickness, but it can also spread by coughing or sneezing. In times of pandemics such as COVID-19, it is almost impossible to collect data manually to help in monitoring disease among population. Because of access to social media by a larger population, opinion mining using sentiment analysis is a powerful tool that will help build models that can detect and predict disease dynamics in a population using re-al-time data [12]. Classification of tweets using combination of different versions of Naïve Bayes and n-gram was utilized in this study. The aim of this research therefore is to use machine learning to classify the spatial distribution of positive and negative sentiments concerning COVID-19, to evaluate the performance of different Naive Bayes machine learning algorithms using n-gram approach in classification of COVID-19 sentiments and to use novel machine learning technique to predict the spatial distribution of COVID-19 sentiments.

2. Methods

2.1. Data Collection and Preprocessing

The data for this study was gathered using the Twitter Streaming API and R. Twitter offers two APIs: a stream API and a Representational State Transfer (REST) API. The streaming API facilitates long-term connections and gives data in real time, which is the fundamental distinction between the two APIs. REST APIs provide for transient and restricted connections, meaning that a set quantity of data can be downloaded per day. Stream API was chosen in this study since it gives access to data as it is being tweeted. The data was acquired using the streamR component in the R application, which was used to collect tweets about the Covid-19 epidemic. The streamR package includes functions that give R users access to the Twitter streaming API as well as a program that parses and transforms collected tweets into R data frames for analysis. The data was streamed from twitter on 14th April 2020 at 16:43:09 to 15th April 2020 at 23:50:53 East African Time, and on 17th April 2020 at 18:24:25 to 18th April 2020 at 16:41:16 East African Time. Tweets with the terms Corona, Covid-19, sanitizer, virus, lockdown, quarantine, and social distance were streamed. The streaming was separated into 2 - 3-hour intervals with roughly 2 seconds delay between each interval to get smaller amounts of streamed tweets. The tweet files were processed and combined into a single excel spreadsheet. The geo-tagged tweets were collected worldwide. The spatial distribution of tweets gathered for this study is shown by Figure 1.

After collecting data, it was pre-processed by Building a corpus variable called corpus and tokenizing text. Then data cleaning was carried out to remove numbers, punctuation, hashtags, urls, annotation @, and retweets RT and removing white spaces. The data was then split into train and test data sets in the ration 80:20. To convert text data into numerical matrices, various methods such as CountVectorizer and Term frequency-Inverse document frequency (TF-IDF)

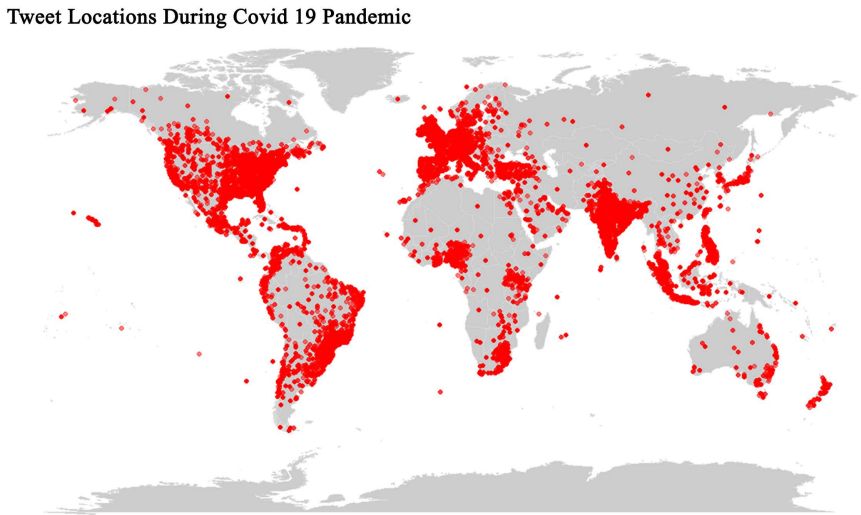

Figure 1. Spatial distribution of tweets.

are utilized. CountVectorizer, according to [13], converts a text document collection into an integer matrix. This method can help you make a sparse count matrix. It also allows text data to be pre-processed before being converted into a vector representation. The term frequency-inverse document frequency (TF-IDF) was coined by [13] to describe the importance of a phrase in a corpus or collection. As the frequency of a specific term in the document increases, so does the TF-IDF value. To control the generality of more common words, the term frequency is offset by the frequency of terms in the corpus. The number of times a term appears in the text is its frequency. The number of times a term appears in all documents is counted using inverse document frequency.

2.2. Machine Learning Methods

Machine Learning Approach algorithms can solve Sentiment Analysis as a standard text classification problem using linguistic features. Under ML approach, text classification methods are divided into supervised and unsupervised learning methods [10]. This study used different versions Naive Bayes machine learning algorithms together using n-gram approach in the classification of the training data; Multinomial Naïve Bayes, Bernoulli Naïve Bayes and Gaussian Naïve Bayes with Unigram, Bigram and Trigram models. The best performing algorithm in the classification was selected for prediction.

Naïve Bayes classification model computes the posterior probability of a class, based on the distribution of the words in a sentence. The model works with the BOWs feature extraction which ignores the position of the word in a sentence. It uses Bayes Theorem to predict the probability that the given preprocessed words belong to a either positive or negative tweets [14].

$$P(\text{label} \mid \text{features}) = \frac{P(\text{features} \mid \text{label}) P(\text{label})}{P(\text{features})} \tag{1}$$

Under Naive assumption, we can rewrite equation (1) as;

$$P(\text{label} \mid f_1, \cdots, f_n) = \frac{p(f_1 \mid \text{label}) p(f_2 \mid \text{label}) \cdots P(f_n \mid \text{label}) P(\text{label})}{P(f_1) P(f_2) \cdots P(f_n)} \tag{2}$$

Multinomial Naïve Bayes distribution is the most appropriate when classifying tweets where events represent a word's occurrence in a single document, [15]. Given p_i as the probability that a tweet is either +ve/−ve and histogram represented by feature vector $x = (x_1, x_2, \cdots, x_n)$ with x_i counting the number of times word *i* occurs in an instance, the likelihood of observing a histogram *x* given classes C_b is expressed as:

$$p(x \mid C_b) = \frac{\left(\sum_i x_i\right)!}{\prod_i x_i!} \prod_i p_{bi}^{x_i} \tag{3}$$

The multinomial naive Bayes classifier becomes a linear classifier when expressed in log-space:

$$\log_{10} p(C_b \mid x) \propto \log_{10} \left(p(C_b) \prod_{i=1}^{n} p_{bi}^{x_i} \right) \tag{4}$$

$$= \log_{10} p(C_b) + \sum_{i=1}^{n} x_i \cdot \log_{10} p_{bi} \tag{5}$$

$$= b + w_b^{\mathrm{T}} x \tag{6}$$

where $b = \log_{10} p(C_b)$ and $w_{bi} = \log_{10} p_{bi}$

Bernoulli Naive Bayes Classifier (BNBC) is a Naive Bayes version that represents a text using a feature vector with binary elements that have the value 1 if the related feature is present and 0 if it is not.

Let

$$Pr(x_t \mid C) \tag{7}$$

be the probability of feature x_t being present in a document of class C and

$$1 - Pr(x_t \mid C) \tag{8}$$

be the probability of feature x_t not being present.

If we again assume independence between features, then we can write the document likelihood as:

$$Pr(X \mid C) \approx \prod_{t=1}^{|X|} Pr(x_t \mid C)^b * \left(1 - Pr(x_t \mid C)^{1-b}\right) \tag{9}$$

where b is the t^{th} value of the feature vector.

The maximum-likelihood estimate that a specific word x_t occurs in class C in text classification is written as:

$$Pr(x_t \mid C) = \frac{dx_t + 1}{dc + 2} \tag{10}$$

where:

- dx_t is the number of documents in the training data set that include the feature x_t and are classified as C.
- dc is the number of documents from class C in the training data set.
- The Laplace smoothing parameters are +1 and +2. These are used to avoid probabilities of 0 or 1 in the case of 0 occurrence of a word within a certain class or 0 occurrence of a specific class in the training data.

Gaussian Naive Bayes is optimal when working with continuous data, one common assumption is that the continuous values associated with each class follow a normal (or Gaussian) distribution.

Let μ_b be the mean values in x associated with class C_b, and let σ_b^2 be the variance of values in x associated with class C_b.

Suppose some observation value of z have been collected, then the probability of z given class C_b, $p(x = z \mid C_b)$ can be calculated by plugging z into normal distribution equation parametrized by μ_b and σ_b^2 giving the features' likelihood as:

$$p(x = z \mid C_b) = \frac{1}{\sqrt{2\pi\sigma_b^2}} \mathrm{e}^{-\frac{(z-\mu_b)^2}{2\sigma_b^2}} \tag{11}$$

To build a simple model, we assume that the data is characterized by a Gaus-

sian distribution with no covariance (independent dimensions) between the parameters. This model can be fitted by simply calculating the mean and standard deviation of the points within each label. The z-score distance between each data point and each class mean is determined, which is the distance from the class mean divided by the class standard deviation.

2.3. N-Gram (Language Models)

N-Gram model is a probabilistic language model used to predict next item in a sequence in the form (n − 1) order Markov. It relies on the Markov assumption: the probability of a word depends only on the previous word without looking too far into the past. N-grams are consecutive sequences of tokens, where the tokens are either words or characters. Under n-grams, when n is 1 then we have traditional bag of wards called unigrams which Naive Bayes uses [16]. This study employed n-grams with sizes greater than 1 to compare with Naive Bayes.

N-gram used to reintroduce some of the lost information in the form of context when using Naive Bayes in form of a short history.

$$P\left(w_i \mid w_1, \cdots, w_{i-1}\right) = P\left(w_i \mid w_{i-1+n}, \cdots, w_{i-1}\right) \tag{12}$$

The maximum likelihood estimates of n-gram probabilities from a corpus expressed as;

$$P\left(w_i \mid w_{i-n+1}, \cdots, w_{i-1}\right) = \frac{count\left(w_{i-n+1}, \cdots, w_i\right)}{count\left(w_{i-n+1}, \cdots, w_{i-1}\right)} \tag{13}$$

In this research, unigram, bigram and trigram models were compared. Under unigram, the assumption is that each word is independent and thus we find the probability of a sequence using;

$$P\left(w_1, w_2, \cdots, w_n\right) = \pi_i P\left(w_i\right) \tag{14}$$

Under the bigram model, the assumption is that each word is independent of its previous word.

$$P\left(w_i \mid w_1, w_2, \cdots, w_{i-1}\right) \approx P\left(w_i \mid w_{i-1}\right) \tag{15}$$

The trigram model assumes that each word is independent of its previous two words:

$$P\left(w_i \mid w_1, w_2, \cdots, w_{i-1}\right) \approx P\left(w_i \mid w_{i-2}, w_{i-1}\right) \tag{16}$$

The MLE estimates for the parameters of an n-gram model MLE for Unigram:

$$P\left(w_i\right) = \frac{C\left(w_i\right)}{N} \tag{17}$$

MLE for Bigram is given by:

$$P\left(w_i, w_j\right) = \frac{count\left(w_i, w_j\right)}{N} P\left(w_i \mid w_j\right) = \frac{P\left(w_i, w_j\right)}{P\left(w_i\right)} = \frac{count\left(w_i, w_j\right)}{\sum_w count\left(w_i, w\right)} \tag{18}$$

$$= \frac{count\left(w_i, w_j\right)}{count\left(w_i\right)} \tag{19}$$

The total number of words is N, the count is C, and the words are w_i and w_j.

As a result, the general case for MLE n-gram parameter estimate is as follows:

$$P\left(w_i \mid w_{i-I+1:i-1}\right) = \frac{C\left(w_{i-I+1:i-1} w_i\right)}{C\left(w_{i-I+1:i-1}\right)} \tag{20}$$

3. Results

3.1. Data Visualization

The terms "quarantine", "lockdown", "pandem", "virus", "covid", and "corona" were the most frequent terms as displayed by the word cloud (**Figure 2**).

The emotions that have high frequency during COVID-19 pandemic are fear, trust, sadness and anticipation as shown by **Figure 3**. This is reflected by the high number of negative sentiments classified than the positive, which shows that COVID-19 significantly impacts individuals' psychological conditions which agree with research by [17].

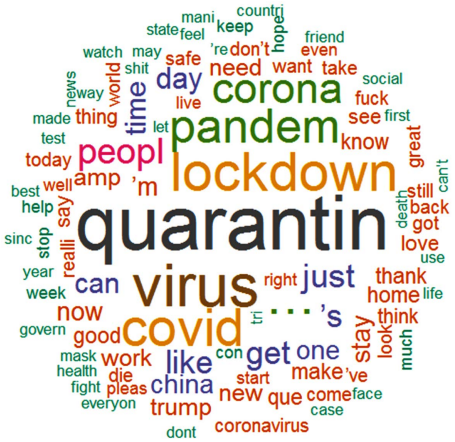

Figure 2. Word cloud.

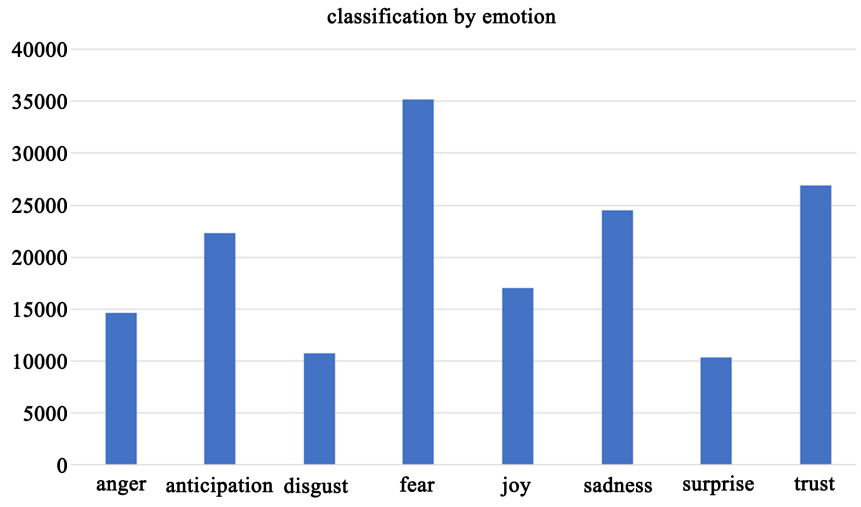

Figure 3. Emotion plot.

3.2. Naïve Bayes Machine Learning Classifiers

In order to evaluate the model classification performance of different versions of Naive Bayes algorithms, we used utilized the training data set to determine the model's classification accuracy. Table 1 below presents classification accuracy for the algorithms used. The accuracy for each classifier was obtained by dividing the number of correctly classified documents (*i.e.* totaling true positive and true negatives) with the total number of documents.

From the accuracy table of the three Naive Bayes classifiers, we see that under Multinomial Naive Bayes classifier, the Bi-Gram Multinomial Naive Bayes algorithm have higher accuracy of 97.02% in classifying tweets in the training data set. Uni-Gram Multinomial Naive Bayes algorithm achieved accuracy of 92.02% while Tri-Gram Multinomial Naive Bayes algorithm achieved accuracy of 94.40%. Under the Bernoulli Naive Bayes classifier, Bigram achieved the highest accuracy of 96.80% followed by Trigram with accuracy of 94.94% then finally Unigram with classification accuracy of 89.34%. Under the Gaussian Naive Bayes classifier, Bigram achieved the highest accuracy of 95.67% followed by Trigram with accuracy of 92.89% then finally Unigram with classification accuracy of 91.43%. After comparing the three Naive Bayes classifiers used for classification of training data set, Bi-gram Multinomial Naive Bayes algorithm was selected for prediction because it achieved the highest accuracy (97.02%). The results for each of the classifiers are displayed in Table 1.

3.3. Prediction by Bi-Gram Multinomial Naïve Bayes

After training the Bi-Gram Multinomial Naive Bayes classifier using the Training data set, prediction was done using the test data set. Table 2 shows the

Table 1. Accuracy of the NB classifiers with different N-Grams.

Multinomial Naive Bayes	Accuracy
1-Gram	0.9202
2-Gram	0.9737
3-Gram	0.9440
Bernoulli Naive Bayes	
1-Gram	0.8934
2-Gram	0.9680
3-Gram	0.9490
Gaussian Naive Bayes	
1-Gram	0.9043
2-Gram	0.9567
3-Gram	0.9289

Table 2. Bi-gram MNB classifier prediction statistics.

Classifier Name	Accuracy	Precision	Recall	F Measure
Bi-Gram MNB	0.8592	0.8550	0.8102	0.8320

accuracy, precision, recall, and F1 measure statistics of sentiments prediction using Multinomial Naive Bayes algorithm. Algorithm accuracy was achieved by dividing the number of correctly classified documents (*i.e.* totaling true positive and true negatives) with the total number of documents. Precision was obtained by getting the percentage of tweets tagged as negative which were genuinely negative *i.e.* true positives divided by sum of true positives and false positives. Recall was obtained by getting the percentage of negative tweets that were classified as negative. *i.e.* true positives divided by sum of true positives and false negatives. F1 measures were obtained by getting the weighted harmonic mean of a test's recall and precision.

We can see that prediction accuracy using MNB is 85.92%. Precision recall and F1-Measure stands at 85.50%, 81.02% and 83.20% respectively. In order to improve the accuracy of our prediction model, we changed the method on how the BOW is formed. BOW, which counts the number of times a word appears in the text, was built using CountVectorizer in the above findings. The more a term is used, the more important it is for classification. To increase the accuracy, the Term Frequency-Inverse Document Frequency (TF-IDF) method was applied, which takes into consideration the product of term frequency and inverse document frequency. The acquired findings are shown in **Table 3**.

When TF-IDF is employed instead of CountVectorizer, the accuracy of the model increased from 85.92% to 87.06%. According to the research by [18], when TF-IDF is used, it achieves higher classification accuracy than when CountVectorizer is used. Using test data, the Bigram MNB algorithm correctly predicted 87.06 percent of tweets. This means that with an accuracy of 87.06%, the Bigram MNB algorithm can predict whether a new tweet would be positive or negative. For Bigram MNB classification, the standard Precision and Recall values achieved are 87% and 83.05%, respectively. This means that our prediction model has an accuracy level of 87 percent exactness and 83.05 percent completeness based on the Multinomial Naive Bayes method for test data. Because the F1-measure is the weighted average of Precision and Recall, it indicates a good model with an F1 score of 84.98%.

After getting the predicted tweets, we plotted the spatial distribution of tweets on the world map and the results are displayed by **Figure 4**.

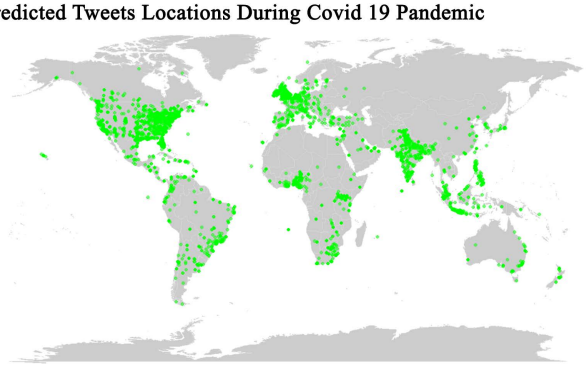

Figure 4. Spatial distribution of predicted tweets.

Table 3. Improved bigram MNB classifier statistics.

Classifier Name	Accuracy	Precision	Recall	F Measure
Bi-Gram MNB	0.8706	0.8700	0.8305	0.8498

4. Conclusions and Recommendations

This study has proven that Machine Language approaches can be used in disease prediction and prevention. Since the usage of social media has grown at an exponential rate, sentiment analysis has become increasingly crucial in extracting people's thoughts, which will help governments make judgments about disease control, particularly during pandemics when physical data collection is impossible. Multinomial Naive Bayes, Bernoulli Naive Bayes, and Gaussian Naive Bayes with n-gram technique, which were used in this study, are effective in sentiment analysis, where people's attitudes were categorized as positive or negative. Because of higher classification accuracy, Multinomial Naive Bayes with bigram was used to predict the spatial distribution of Covid-19 feelings as either positive or negative. N-gram model plays an important role in relaxing the Naive Bayes assumption by utilizing Markov assumption by finding the likelihood of a future unit without having to look too far into the past.

Twitter has proven to be a valuable resource for classification and disease prediction. Twitter data is real-time and accessible via API from a big number of users in various geographic regions. This research contributes to the surveillance system in order to better understand the changing scenario around the Covid-19 pandemic, such as mental illness, job loss, and government involvement (lockdown, wearing masks etc.). The patterns and emotions discovered in public tweets could be used to guide specific intervention initiatives aimed at resolving the problem. Because of the large number of tweets as the disease spread, instances and a potential epidemic of Covid-19 could be identified early enough, implying that the Twitter community understood the disease's severity. This is an excellent opportunity to encourage the people to take action, to take preventative steps as soon as possible. The public and authorities may be better able to respond to the spread of the disease if they can quickly detect and use social media postings to alert them to the situation.

Recommendations

Other research can explore other methods of text classification such as lexicon-based approach in classifying and modeling disease outcome. This study can be extended to model other pandemic outbreaks and discover sentiment emotion in similar fields. Future studies can focus on other social media platforms such as Facebook, Instagram and snap chat in modeling disease outcome and compare outcome. Additionally, future studies can include tweets from other languages apart from English such as Italian, Germany, and Spanish to classify and predict disease outcome.

Acknowledgements

I would like to thank Professor Mwalili Samuel and Dr. Okango Elphas for their professional mentorship, leadership and guidance, and also my wife for her moral support and unique quality attitude.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Samuel, J., Ali, G.G., Rahman, M., Esawi, E. and Samuel, Y. (2020) Covid-19 Public Sentiment Insights and Machine Learning for Tweets Classification. *Information*, **11**, 314. https://doi.org/10.3390/info11060314

[2] Ivanov, D. (2020) Predicting the Impacts of Epidemic Outbreaks on Global Supply Chains: A Simulation-Based Analysis on the Coronavirus Outbreak (COVID-19/SARS-CoV-2) Case. *Transportation Research Part E*: *Logistics and Transportation Review*, **136**, Article ID: 101922. https://doi.org/10.1016/j.tre.2020.101922

[3] Dicker, R.C., Coronado, F., Koo, D. and Parrish, R.G. (2006) Principles of Epidemiology in Public Health Practice; an Introduction to Applied Epidemiology and Biostatistics.

[4] Jin, D., Jin, Z., Zhou, J.T. and Szolovits, P. (2019) Is Bert Really Robust? Natural Language Attack on Text Classification and Entailment.

[5] Mäntylä, M.V., Graziotin, D. and Kuutila, M. (2018) The Evolution of Sentiment Analysis—A Review of Research Topics, Venues, and Top Cited Papers. *Computer Science Review*, **27**, 16-32. https://doi.org/10.1016/j.cosrev.2017.10.002

[6] Adhikari, N.C.D., Alka, A. and Garg, R. (2017) HPPS: Heart Problem Prediction System Using Machine Learning. *CS & IT Conference Proceedings*, Vol. 7, 23-37. https://doi.org/10.5121/csit.2017.71803

[7] Zhao, J., Liu, K. and Xu, L. (2016) Sentiment Analysis: Mining Opinions, Sentiments, and Emotions. Cambridge University Press, Cambridge. https://doi.org/10.1162/COLI_r_00259

[8] Prabhakar Kaila, D. and Prasad, D.A. (2020) Informational Flow on Twitter—Corona Virus Outbreak-Topic Modelling Approach. *International Journal of Advanced Research in Engineering and Technology*, **11**, 128-134.

[9] Medford, R.J., Saleh, S.N., Sumarsono, A., Perl, T.M. and Lehmann, C.U. (2020) An "Infodemic": Leveraging High-Volume Twitter Data to Understand Public Sentiment for the COVID-19 Outbreak. https://doi.org/10.1101/2020.04.03.20052936

[10] Suppala, K. and Rao, N. (2019) Sentiment Analysis Using Naïve Bayes Classifier. *International Journal of Innovative Technology and Exploring Engineering*, **8**, 264-269.

[11] Dubey, A.D. (2020) Twitter Sentiment Analysis during COVID19 Outbreak. https://doi.org/10.2139/ssrn.3572023

[12] Agarwal, A., Xie, B., Vovsha, I., Rambow, O. and Passonneau, R.J. (2011) Sentiment Analysis of Twitter Data. *Proceedings of the Workshop on Language in Social Media*, Portland, 23 June 2011, 30-38.

[13] Garreta, R. and Moncecchi, G. (2013) Learning Scikit-Learn: Machine Learning in Python. Packt Publishing Ltd., Birmingham.

[14] Dey, L., Chakraborty, S., Biswas, A., Bose, B. and Tiwari, S. (2016) Sentiment Anal-

ysis of Review Datasets Using Naive Bayes and k-nn Classifier.

[15] Manning, C.D., Schütze, H. and Raghavan, P. (2008) Introduction to Information Retrieval. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511809071

[16] Hiemstra, D. (2001) Using Language Models for Information Retrieval. Taaluitgeverij Neslia Paniculata, Enschede.

[17] Browning, M.H., Larson, L.R., Sharaievska, I., Rigolon, A., McAnirlin, O., Mullenbach, L., Alvarez, H.O., *et al.* (2021) Psychological Impacts from COVID-19 among University Students: Risk Factors across Seven States in the United States. *PLoS ONE*, **16**, e0245327. https://doi.org/10.1371/journal.pone.0245327

[18] Manish, S. (2020) Sentiment Analysis: An Introduction to Naive Bayes Algorithm.