


Sentiment Analysis on Worldwide COVID-19 Outbreak



Rakshatha Vasudev , Prathamesh Dahikar, Anshul Jain, and Nagamma Patil

Abstract Sentiment analysis has proved to be an effective way to easily mine public opinions on issues, products, policies, etc. One of the ways this is achieved is by extracting social media content. Data extracted from the social media has proven time and again to be the most powerful source material for sentiment analysis tasks. Twitter, which is widely used by the general public to express their concerns over daily affairs, can be the strongest tool to provide data for such analysis. In this paper, we intend to use the tweets posted regarding the COVID-19 pandemic for a sentiment analysis study and sentiment classification using BERT model. Due to its transformer architecture and bidirectional approach, this deep learning model can be easily preferred as the best choice for our study. As expected, the model performed very well in all the considered classification metrics and achieved an overall accuracy of 92%.

Keywords COVID-19 · Sentiment analysis · Opinion mining · Word embedding · BERT · Classification · Fine tuning

1 Introduction

Machine Learning is the science of making computers act without being explicitly programmed. Machine Learning has quite gained its importance in the past decade and its popularity is only going to increase in the near future. All of us use machine learning regularly without realizing and it is very useful in making decisions that are to classify a given data. Sentiment analysis is a part of machine learning study which helps us to analyze the sentiment of a piece of text. We can use sentiment analysis for classification tasks. The use of sentiment analysis on social media data to get public opinion is widely accredited. It helps in processing huge amounts of data in order to

R. Vasudev (✉) · P. Dahikar · A. Jain · N. Patil
National Institute of Technology Karnataka, Mangaluru, India
e-mail: rakshathavasudev@gmail.com

N. Patil
e-mail: nagammapatil@nitk.edu.in

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2024
M. D. Borah et al. (eds.), *Big Data, Machine Learning, and Applications*, Lecture Notes in Electrical Engineering 1053, https://doi.org/10.1007/978-981-99-3481-2_47

get the sentiments or opinions of the people about a context. Traditional sentiment analysis can miss out on highly valued insights. The advancements in deep learning can provide us with sophisticated models to classify the data being used for sentiment analysis by providing them with contextual meaning. For our study, we have used the BERT (Bidirectional Encoder Representations from Transformers) [17] model for classification of tweets into their sentiments which are represented by the three class labels: positive (denoted by 0), negative (denoted by 1) and neutral (denoted by 2). We have also used word clouds of tweets to plot the most frequently used terms in the tweets. These plots give us an accurate visual representation of the most prominently used words in the tweets. These representations can help create an awareness.

2 Literature Survey

Ji et al. [1] address the issue of spreading public concern about epidemics. They have used Twitter messages, trained and tested them on three Machine Learning Models, namely Naive Bayes (NB), Multinomial Naive Bayes (MNB) and Support Vector Machine (SVM) to obtain the best results. Alessa et al. [2] have reviewed the existing solutions that track the influenza flu outbreak in real time, with the use of weblogs and social networking sites. The paper concludes that social networking sites can provide better predictions when used to conduct real-time analysis. Adhikari et al. [3] have combined word embeddings, Term Frequency-Inverse Document Frequency (TF-IDF) and word-n grams with various algorithms for data mining and deep learning such as Support Vector Machine, NB and RNN-LSTM. Naive Bayes along with Term Frequency-Inverse Document Frequency performed the best compared to other methods used. Rastogi et al. [4] used decomposition (Normalization Form and Compatibility decomposition) for preprocessing and used NLTK (Natural Language Toolkit) package for tokenization and Twitter preprocessor to remove tags, Hashtags, Reserved words, URLs and mentions. TF-IDF and bag of words were used to find the most frequent words in the corpus. VADER (Valence Aware Dictionary and sentiment reasoner) was used for sentiment analysis which also takes care of emojis. For classification, this paper used SVM and BERT model. Pokharel et al. [5] have used Tweepy Python library for data collection. Necessary fields are scraped and the TextBlob is used for checking the polarity of the tweet (positive, negative or neutral). Singh et al. [6] have used artificial intelligence (AI) techniques for prediction of epidemic outbreak. Two approaches have been used in this paper, one is the societal approach and the other is the epidemiology approach. The societal approach includes analyzing the public awareness about the epidemic using the collected tweets and then performing sentiment analysis on them. The computational epidemiology approach includes analysis and prediction of future trends based on medical datasets. Kabir and Madria [7] have built up a real-time web application for COVID-19 Tweets Data Analyzer. They have collected the data from March 5, 2020 and kept fetching the tweets using the tweepy package of Python. In this paper, authors have performed sentiment analysis on the topics related to trend-

ing topics to understand the sentiment of human emotions. They also provide clean dataset named coronaVis Twitter dataset based on United States. Nemes et al. [8] have analyzed the signs and sentiments of the Twitter users based on the main trends using NLP (Natural Language Processing) and Sentiment Analysis with RNN(Recurrent Neural Network). The trained model worked much more accurately with a very high accuracy in determining the emotional polarity of tweets (including ambiguous ones). Wang et al. [9] proposed fine tuned BERT model for the classification of the sentiments of posts and have used tf-idf to extract topics of posts with different sentiments. Negative sentiments of the posts are used to predict the epidemic. Based on our survey, we have concluded that the existing models struggle to evaluate the language complexities like double negatives, words with multiple meanings and context-free representations. Also, the models require a huge set of training data for sentiment analysis. Our work focuses on conducting a sentiment analysis to help people to make informed decisions by knowing what is happening around the globe and also to develop a sentiment classification model which is highly performance enhanced with limited data regarding COVID-19.

3 Architecture

The Twitter data collected for sentiment analysis is analyzed and allotted class labels using their sentiments. The data is also analyzed for creating word clouds of location and most frequently used words from tweets. The class labels are analyzed to get an idea of the distribution of sentiments of the tweets. The tweets are preprocessed to remove punctuation, stop words and other unnecessary data. The data is then used for sentiment classification using BERT (Bidirectional Encoder Representations from Transformers). The model performance is evaluated using classification metrics (see Fig. 1).

4 Methodology

4.1 Data Collection and Analysis

The dataset for the sentiment analysis was obtained from Kaggle. It consisted of about 170k tweets from all over the world about COVID-19. The data frame ultimately prepared consisted of tweet id and tweets extracted from the dataset for our analysis purpose. 100 recent tweets specific to India were also analyzed using word clouds. The tweets in the dataframe were assigned class labels using Textblob to make this a supervised learning problem. Textblob, a Python library, is widely used for various textual data processing tasks. One of them is sentiment analysis from texts. The sentiment of the textblob object is returned in a tuple which consists of polarity

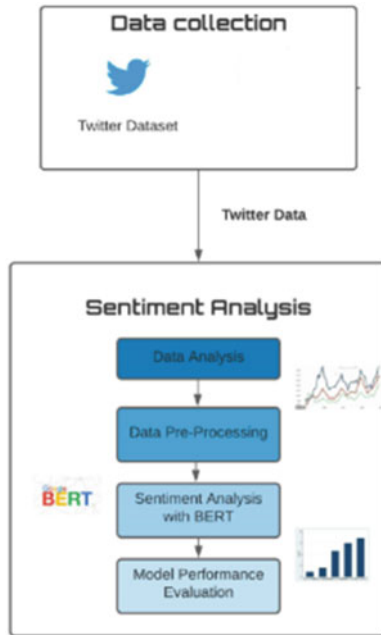


Fig. 1 Workflow/design of the proposed model

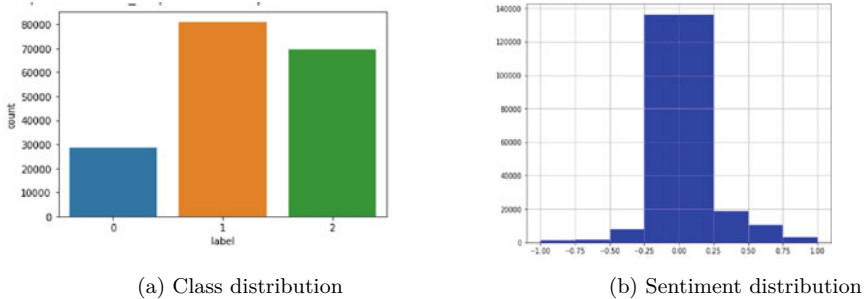


Fig. 2 Data distribution

and subjectivity. The polarity property is considered for class labels generation. The range of polarity values is $[-1, 1]$. If the polarity of a tweet was less than 0 it was assigned a class label 0 (negative tweet), else if the polarity was equal to 0 it was assigned 1 (neutral tweet) else the class label was 2 (positive tweet). From Fig. 2a, we can see that the data has imbalanced class distribution. The number of negative tweets is little lower than the other two. But this problem can be handled by using metrics which will evaluate the model class wise. The sentiments distribution of the tweets was plotted, and from Figure 2b, we can see that the majority of the tweets are distributed among neutral and positive sentiments.

4.2 Data Pre-processing

The Twitter data was preprocessed using a message cleaning pipeline to remove unnecessary data. The punctuations from the tweets were removed using string.punctuation. All the stopwords were removed using nltk’s stopwords list. And finally all the video and hyperlinks were removed using Python regex. The pre-processed data is shown in Fig. 3.

Fig. 3 Pre-processed data

| id | Location | tweets | label | category |
|----|---------------------|---|-------|----------|
| 0 | astroworld | smelled scent hand sanitizers today someone pa... | 0 | Negative |
| 1 | New York, NY | Hey Yankees YankeesPR MLB wouldnt made sense p... | 2 | Positive |
| 2 | Pewee Valley, KY | diane3443 wdunlap realDonaldTrump Trump never ... | 1 | Nuetral |
| 3 | Stuck in the Middle | brookbanktv one gift COVID19 give appreciation... | 1 | Nuetral |
| 4 | Jammu and Kashmir | 25 July Media Bulletin Novel CoronaVirusUpdate... | 1 | Nuetral |

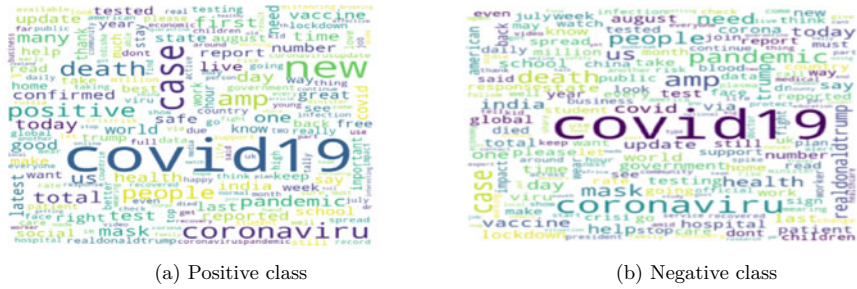
4.3 World Cloud Analysis

A word cloud of locations of the tweets was plotted for each class to analyze the severity and pattern of the epidemic is shown in Fig. 4. We can see that countries like India, United States, South Africa, etc. are where the most tweets are from indicating the Twitter activity of people from these countries is very high. This also means that people from these regions are very concerned over the situation.

Also, the most frequently used words from each class were plotted in the word cloud. This gives us an idea of how people are reacting to the epidemic and their sentiments toward it. It can also give us important information on precautions to be taken at the earliest in case of a pandemic in regions where it may not have still affected. We can see from Fig. 5a the most frequent words in positive tweets. Words

Fig. 4 Wordcloud of locations of tweets-Positive class





(a) Positive class

(b) Negative class

Fig. 5 Word cloud of frequently used words in tweets



(a) Positive class

(b) Negative class

Fig. 6 Word cloud of frequently used words in latest tweets specific to India

like good, safe, great and vaccine tells us that people are trying to be positive minded, have concerns over vaccine and to stay healthy during the pandemic. It also raises concerns over wearing masks, schools opening, lockdown, etc. We can also see words such as tested meaning people are aware of the importance of testing and are getting tested. In Fig. 5b, we can see words like government, country, etc. giving us an idea that people are expressing views on the government’s action toward pandemic. We can derive such information from the frequently used words using word cloud.

We have also considered the latest tweets from India for the word clouds. From Fig. 6a, we can see words like vaccination, happy, great, fun roll, etc. indicating the situation might be under control. In Fig. 6b, we can see words like new cases, active and variant expressing concerns over new variants that might be spreading.

4.4 Sentiment Classification Using Bert

We propose to use BERT(Bidirectional Encoder Representations from Transformers) to train our model which will classify the tweets into their sentiments. The reasons for using BERT for this classification task was

- Sentiment classification tasks always require a huge set of data for model training. Since BERT is already pre-trained with billions of data from the web, it eliminates the need for a huge dataset for model training. Fine tuning gives the desired results for our classification.
- It works in two directions simultaneously (bidirectional). Other language models look for the context of the word either from left or right. But BERT is bidirectionally trained which means the words can have deeper context hence the classification task performance can be improved.

4.4.1 Fine Tuning BERT for Classification

The preprocessed dataset which consisted of 1,00,439 tweets was used for the BERT model. The dataset was divided into train and validation set using train-test-split with 20% test size as shown in Fig. 7. Here, the dataset for classification will consist only of the tweets and the label columns.

BERT base uncased tokenizer from Hugging Face’s transformers library which has 12-layer, 768-hidden-nodes, 12- attention-heads and 110 Million parameters was used for tokenization of the tweets. Once the tweets are encoded using the tokenizer, we can get the input features for BERT model training which are Input id and attention masks(both of which we can get from the encoded data). Input id are the integer sequences of the sentences. Attention masks are a list of binary numbers 0s and 1s representing which tokens to be given attention by the model and for which it should not. There’s also one more input feature required by the BERT model which is labels. All the integer input features(both the train and validation sets) should be converted to tensor datasets. The tensor datasets will be used to get train and validation data loaders. Optimizers (AdamW) and schedulers (linear-schedulewith-warmup) are defined to control the learning rate through the epochs. BERT base uncased BertForSequenceClassification model from the transformers library is defined for training. In training mode, the model will be trained batch wise with the input features. The loss obtained from the outputs is backward propagated and other parameters(optimizers and schedulers) are updated. At the end of each epoch, the validation data loader is evaluated in model evaluating mode. This also returns the validation loss, predictions and true values. So at the end of each epoch,

Fig. 7 Train and validation sets

| category | label | data_type | tweets |
|----------|-------|-----------|--------|
| Negative | 0 | train | 13382 |
| | | val | 3346 |
| Nuutral | 1 | train | 33318 |
| | | val | 8329 |
| Positive | 2 | train | 33651 |
| | | val | 8413 |

we can analyze training and validation losses. The model is saved using torch. The saved model is to be loaded with the same parameters as of the model defined for all the keys to be perfectly matched. Once the model is loaded, the performance can be evaluated by various metrics using the validation set data loaders.

4.5 Evaluation Metrics

The metrics used for evaluating the classification task were as follows: Accuracy, Precision, Recall and F1- score. All these metrics are based on the confusion matrix.

- True Positives (TP): The cases which are predicted positive and are actually positive.
- True Negatives (TN): The cases which are predicted negative and are actually negative.
- False Positives (FP): The cases which are predicted positive but are negative.
- False Negatives (FN): The cases which are predicted negative but are positive.

4.5.1 Precision

Precision represents what percentage of predicted positives are actually positive and can be calculated easily by

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \quad (1)$$

4.5.2 Recall

Recall represents what percentage of actual positives are predicted correctly and calculated by

$$\text{Recall} = (\text{TP})/(\text{TP} + \text{FN}) \quad (2)$$

4.5.3 F1-Score

F1-score is the measure of accuracy of a model on the dataset and it is the harmonic mean of precision and recall.

$$\text{F1 - score} = (2 * \text{Precision} * \text{Recall})/(\text{Precision} + \text{Recall}) \quad (3)$$

4.5.4 Accuracy

Accuracy represents how often is our classifier correct, i.e., it is defined as the percentage of predictions that are predicted correctly and is calculated by the following formula:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FN} + \text{FP}) \quad (4)$$

4.5.5 Macro average

Macro average is the arithmetic mean of all the values irrespective of proportion of each label in the dataset. For example, if we want to find the macro-average precision of n classes, individual precisions being $p_1, p_2, p_3, \dots, p_n$ then macro-average precision (MAP) is the arithmetic mean of all these.

$$\text{MAP} = (p_1 + p_2 + p_3 + \dots + p_n) / n \quad (5)$$

4.5.6 Weighted Average

Weighted average is the average with proportion of each label in the dataset. Some weights are assigned to each label based on proportion of each label in the dataset. For example, if we want weighted-average precision of n classes, precisions being $p_1, p_2, p_3, \dots, p_n$ and assigned weights being $w_1, w_2, w_3, \dots, w_n$ then weighted-average precision (WAP) can be calculated as

$$\text{WAP} = (p_1 * w_1 + p_2 * w_2 + \dots + p_n * w_n) / n \quad (6)$$

Table 1 Metrics evaluated class wise

| Classes | Accuracy | Precision | Recall | F1-score |
|----------|----------|-----------|--------|----------|
| Positive | 0.974 | 0.96 | 0.90 | 0.93 |
| Neutral | 0.900 | 0.90 | 0.97 | 0.94 |
| Negative | 0.832 | 0.88 | 0.83 | 0.86 |

Table 2 Weighted- and Macro-average results of the metrics

| Macro average | | | Weighted average | | |
|---------------|--------|----------|------------------|--------|----------|
| Precision | Recall | F1-score | Precision | Recall | F1-score |
| 0.91 | 0.90 | 0.91 | 0.92 | 0.92 | 0.92 |

5 Results and Analysis

The BERT model was trained on GPU (CUDA enabled) with 12.72 GB RAM in Google Colab. The model was trained for one epoch and the classification report and class-wise accuracies were evaluated and the same are tabulated in Tables 1 and 2. The classification report (from sklearn. metrics) gives us the class-wise Precision, Recall and F1-score and also the macro and weighted averages of these metrics. We can observe that the weighted-average scores are better than the macro-average scores. It is because the number of neutral and positive classes in our dataset is slightly higher than the negative class. The overall accuracy of the model was 92%. This indicates that our model can be used over time to classify huge amounts of textual data on COVID-19-related issues quite accurately. The model also performed well by achieving desired class-wise accuracy, precision, recall and F1-score. Both positive and neutral classes had values above 0.9 in all these metrics.

6 Conclusion and Future Work

In this paper, we have proposed machine learning-based approaches to sentiment analysis and sentiment classification specifically for the COVID-19 pandemic worldwide. We can analyze the effect of pandemic using the Twitter data and also analyze the response from the people about the epidemic using the word clouds which plot the most frequently used words from the tweets. These word clouds also help us in knowing how different regions are being affected by the pandemic and can create an awareness among people to prevent it. The model was trained with pre-trained BERT for classification. The model performed very well with 92% accuracy, 0.91 macro-average Precision and F1-Score, 0.90 Recall and 0.92 weighted-average Precision, Recall and F1-Score. The use of BERT for sentiment analysis has certainly given the best results and makes our model a reliable one.

Sentiment analysis, though useful, is always subjective. The opinions differ from people to people. Also, it's very difficult to correctly contextualize sentiments such as sarcasm and negations. Further, this project can be automated to continuously fetch tweets to feed as data for the model while ensuring that there is no overfitting. BERT can further be fine tuned to get better results.

References

1. Ji X, Chun S, Wei Z, Geller J (2015) Twitter sentiment classification for measuring public health concerns. *Soc Netw Anal Min* 5
2. Alessa A, Faezipour M (2018) A review of influenza detection and prediction through social networking sites. *Theor Biol Med Model* 15

3. Adhikari ND, Kurva VK, Suhas S, Kushwaha JK, Nayak AK, Nayak SK, Shaj V (2018) Sentiment classifier and analysis for epidemic prediction. *Comput Sci Inf Technol (CS and IT)*
4. Rastogi N, Keshtkar F (2020) Using BERT and semantic patterns to analyze disease outbreak context over social network data. In: *Proceedings of the 13th international joint conference on biomedical engineering systems and technologies*
5. Pokharel B (2020) Twitter sentiment analysis during COVID-19 outbreak in Nepal. *SSRN Electron J*
6. Singh R, Singh R (2021) Applications of sentiment analysis and machine learning techniques in disease outbreak prediction-a review. *Mater Today Proc*
7. Kabir M, Madria S (2020) CoronaVis: a real-time COVID-19 tweets data analyzer and data repository. <https://arxiv.org/abs/2004.13932>
8. Nemes L, Kiss A (2020) Social media sentiment analysis based on COVID-19. *J Inf Telecommun 5*:1–15
9. Wang T, Lu K, Chow K, Zhu Q (2020) COVID-19 sensing: negative sentiment analysis on social media in China via BERT model. *IEEE Access 8*:138162–138169
10. Siriyasatien P, Chadsuthi S, Jampachaisri K, Kesorn K (2018) Dengue epidemics prediction: a survey of the state-of-the-art based on data science processes. *IEEE Access 6*:53757–53795
11. Pham Q, Nguyen D, Huynh-The T, Hwang W, Pathirana P (2020) Artificial intelligence (AI) and big data for coronavirus (COVID-19) pandemic: a survey on the state-of-the-arts. *IEEE Access 8*:130820–130839
12. Manguri K, Ramadhan R, Amin P. *Kurd J Appl Res*. <http://doi.org/10.24017/kjar>
13. Agarwal A, Xie B, Vovsha I, Rambow O, Passanneau R (2011) Sentiment analysis of Twitter data. In: *Proceedings of the workshop on languages in social media*, pp 30–38
14. Rajput N, Grover B, Rathi V (2020) Word frequency and sentiment analysis of twitter messages during Coronavirus pandemic. <https://arxiv.org/abs/2004.03925>
15. Kruspe A, Häberle M, Kuhn I, Zhu X (2020) Cross-language sentiment analysis of European Twitter messages during the COVID-19 pandemic. <https://aclanthology.org/2020.nlpCOVID19-acl.14>
16. Boon-Itt S, Skunkan Y (2020) Public perception of the COVID-19 pandemic on twitter: sentiment analysis and topic modeling study. *JMIR Public Health Surveill 6*:e21978
17. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. <https://arxiv.org/abs/1810.04805>