Unified approach to discover Sentiment analysis of Covid-19 Twitter Data utilizing Machine Learning Classifiers

Sudeep Kisan Hase[1]
Research Scholar
Department of Computer Science and Enginering
Oriental University, Indore (India)
hase.sudeep@gmail.com

Dr. Rashmi Soni[2,]
Professor, Dayananda Sagar Academy of Technology & Management, Bangaluru
Research Supervisor, Oriental University, Indore (India) drrashmiofficial@gmail.com

ABSTRACT

Sentiment analysis has been emerging factor from Covid-19 wave. Finding out polarity of data cloud is not enough. Human emotions always give better idea about behavioral characteristics. Machine learning classifiers and its result surely gives impactful idea about specific condition. This chapter will give comparative and unified approach among machine learning classifiers.

Keywords—ML, Sentiment Analysis, NLP, Classifiers

I. INTRODUCTION

COVID-19 vaccines have brought much relief and newfound optimism to so many after a long time of sickness, devastation, grief, and hopelessness. Every day, news stories and Twitter spheres are filled with discussions about vaccination progress, accessibility, efficacy, and side effects. In spite of this, as online users, our visibility is very limited to the echo chambers that we create within ourselves. Hence, this chapter was motivated by a desire to increase my understanding of the global pandemic through Twitter data[2].

Since its first discovery in the Chinese town of Wuhan in December 2019, the highly contagious coronavirus disease (COVID-19) has been transmitted to 212 countries and territories, influencing tens of millions of people. The disease was identified for the first time in a student travelling from Wuhan on the last day of January in 2020 in India, a country with a population of over 1.3 billion people[6].

COVID-19 disease and vaccines have been the subject of a lot of tweets, making it nearly impossible for a human to read through it all. Thus, the urge to better understand the global epidemic using Twitter data was the driving force behind this initiative. There have been so many tweets about 19 vaccines that it would take a human being a very long time to read them all. Natural language processing (NLP) allows us to acquire insight into an enormously complicated and broad conversation by examining narrative aspects, doing sentiment analyses, and visualizing word clouds[1][3].

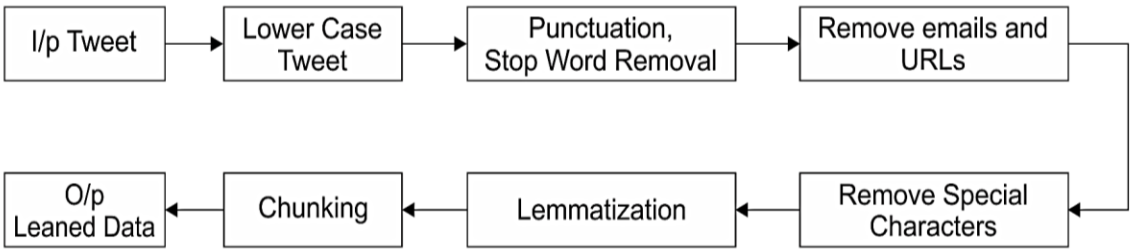


Fig. 1: Sentiment Analysis Preprocessing Steps

The data which is fetched from twitter API has been preprocessed with some steps[4]. Lower case word will required for processing, so all the words are converted into lower case words. Comma, Apostrophe, Hyphen, Ellipsis, Full Stop, Exclamation Mark, Questions Marks, Colon, Bracket, Splash, Quotation mark etc. punctuation and stop words will be removed. The URL links and email IDs should be removed[5][7][8]. Word

should be reduced to its base form. Then Noun phrases from sentence are extracted. At the end we will get output in the forms of words.

Text is scanned, processed, analyzed, and interpreted by a natural language processing system for textual data. The technology first preliminary processes the text via a number of phases to create a format that is more organized. The term "preprocessing stream" refers to a procedure where the outcome from one stage functions as the input for the subsequent one[11][12].

II. LITERATURE SURVEY

Table 1: Different Survey papers and accuracies of classifier

Classifiers used	Accuracy	Data Set Used	References
Support Vector Machine, Naive Bayes, Decision Tree	Accuracy of SVM is around 92%	Mexican Earthquake Data	Ref 1 (2020)
Support Vector Machine, Back Propagation Neural Network	Accuracy of SVM is around 70%	Kaggle dataset unclassified tweets	Ref 2 (2020)
Naive Bayes, Decision Tree, K-Nearest Neighbors, and Support Vector Machine, Recurrent Neural Network	Accuracy of RNN-LSTM is around 94%	IMDB, Airline and Amazon Dataset	Ref 3 (2018)
SVMLiblinear	Accuracy of SVMLiblinear is around 98%	Unclassified twitter data. Weka is used for classification	Ref 4 (2019)
Hybrid Approach	Accuracy of SVM, Random Forest is around 87%	Manual Twitter Dataset	Ref 5 (2021)
Bidirectional Encoder Representations from Transformers (BERT), Long Short-Term Memory Network (LSTM), and Convolutional Neural Network (CNN)	Accuracy of BERT classifier is around 65%	SemEval-2016 dataset for Twitter	Ref 6 (2020)
Bidirectional Encoder Representations from Transformers (BERT), and Valence Aware Dictionary and sEntiment Reasoner (VADER)	Accuracy of SVM around 95%	Covid-19 Twitter Data. Results are carried out on GPU	Ref 7 (2021)
NodeXL, Natural Language Tool Kit (NLTK), and Valence Aware Dictionary and sEntiment Reasoner (VADER)	VADER (Polarity)	US presidential election	Ref 8 (2019)
Support Vector Machine (SVM)	Accuracy of SVM is around 97%	Breast cancer disease data	Ref 9 (2016)
BERT, root mean square error (RMSE)	Accuracy of RMSE is around 93%	Humor data analysis	Ref 10 (2019)
CNN Convolutional Neural Network (CNN)	GPU Parallelism	Manual Twitter Data	Ref 11 (2017)
XgBoost (eXtreme Gradient Boosting)	Accuracy of XgBoost is around 97%	CovidSenti Dataset	Ref 12 (2021)
Long Short-Term Memory Network (LSTM), and Convolutional Neural Network (CNN)	Accuracy of LSTM is around 84%	US airline, IMDB and GOP debate dataset	Ref 13 (2020)
Support Vector Machine,	Accuracy of ULMFit		Ref 14 (2022)

Universal Language Model Fine-tuning (ULMFit SVM)	SVM is around 99%		

III. WORKING

A. Preprocessing

Lowering whole tweets and printing it

```
[5]: df = df.apply(lambda x:str(x).lower())
     for i,j in enumerate(df,1):
         print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car. i got out and this person
laughed at me. i felt such a great anger that i got in my car and went away.

2 she still , after all these years , did not know , and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother .
```

Fig 2 : Step1 - Lowercase String

Applying cont_exp on tweets and printing it

```
[6]: df = df.apply(lambda x: th.cont_exp(x)) #you're -> you are; i'm -> i am
     for i,j in enumerate(df,1):
         print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car. i got out and this person
laughed at me. i felt such a great anger that i got in my car and went away.

2 she still , after all these years , did not know , and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother .
```

Fig 3 : Step2- Counting words

Removing emails if found and removing it

```
[7]: df = df.apply(lambda x: th.remove_emails(x))
     for i,j in enumerate(df,1):
         print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car. i got out and this person
laughed at me. i felt such a great anger that i got in my car and went away.

2 she still , after all these years , did not know , and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother .
```

Fig 4 : Step3 – Deletion of Email IDs

Removing HTML tags if Found

```
[8]: df = df.apply(lambda x: th.remove_html_tags(x))
     for i,j in enumerate(df,1):
         print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car. i got out and this person
laughed at me. i felt such a great anger that i got in my car and went away.

2 she still , after all these years , did not know , and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother .
```

Fig 5 : Step4- Delete Hypertext ML Tags

Removing Special characters if found

```
[9]: df = df.apply(lambda x: th.remove_special_chars(x))
     for i,j in enumerate(df,1):
         print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car i got out and this person
laughed at me i felt such a great anger that i got in my car and went away

2 she still after all these years did not know and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother
```

Fig 6 : Step5 – Deleting Special Characters

Removing accented characters if found

```
[10]: df = df.apply(lambda x: th.remove_accented_chars(x))
      for i,j in enumerate(df,1):
          print(i,j,"\n")
```

```
1 one day in a crossroad somebody crashed my car i got out and this person
laughed at me i felt such a great anger that i got in my car and went away

2 she still after all these years did not know and one hand clenched in
involuntary anguish at what she thought of as her intolerable betrayal of her
brother
```

Fig 7 : Step6- Deleting Western Lang. Accented Words

Translating words into their base form

```
[11]: df = df.apply(lambda x: th.make_base(x)) #ran -> run,
      for i,j in enumerate(df,1):
          print(i,j,"\n")
```

```
1 one day in a crossroad somebody crash my car i get out and this person laugh
at me i feel such a great anger that i get in my car and go away

2 she still after all these year do not know and one hand clench in involuntary
anguish at what she think of as her intolerable betrayal of her brother
```

Fig 8 : Step7 – Returning String to Base Form

Removing stopwords from tweets and printing final preprocessed tweets

```
[12]: def remove_stopwords(x):
          custom_list =['I','i','my','myself','we','our','ours','ourselves','you',
      "you're", "you've", "you'll", "you'd", 'your',
                        'yours', 'yourself','yourselves', 'he', 'him', 'his',
      'himself', 'she', "she's", 'her', 'hers', 'herself',
                        'it', "it's", 'its', 'itself', 'they','them', 'their',
      'theirs', 'themselves', 'this', 'that', "that'll",
                        'these', 'those','am', 'is', 'are', 'was', 'were', 'be',
      'been', 'being', 'have', 'has', 'had', 'having',
                        'do', 'does', 'did', 'doing','a', 'an', 'the', 'and',  's',
      't', 'now', 'd', 'll', 'm', 'o', 're', 've',
                        'y', 'ain','ma']
          tokens = word_tokenize(x)
          sentence_without_stopword = [k for k in tokens if not k in custom_list]
          return ' '.join(sentence_without_stopword)
      df = df.apply(lambda x: remove_stopwords(x))
      for i,j in enumerate(df,1):
          print(i,j,"\n")
```

```
1 one day in crossroad somebody crash car get out person laugh at me feel such
great anger get in car go away

2 still after all year not know one hand clench in involuntary anguish at what
think of as intolerable betrayal of brother
```

Fig 9 : Step8- Deleting Stopwords

These preprocessing steps have cleaned twitter data so that we can perform training and testing phases on it. If these steps are not follow then we will get false result in classifier accuracy.

B. **Experimental Workflow**

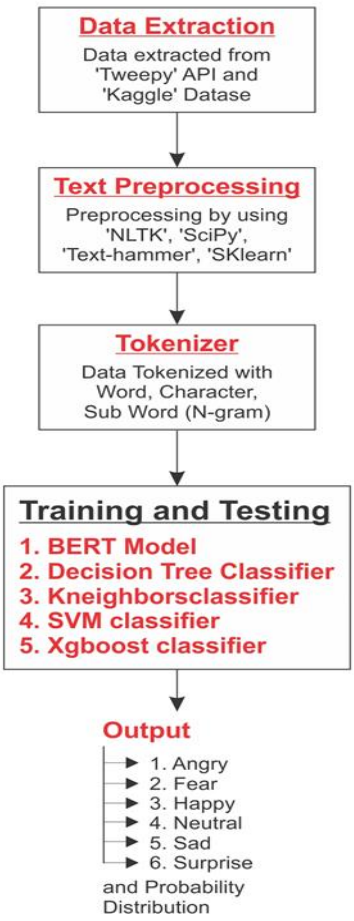


Fig. 10 Experimental Workflow for Covid-19 Twitter Data Sentiment Analysis

The first and foremost process is carrying out data from the Twitter. Tweepy API will give interface to extract data from twitter login. Kaggle provides service of machine learning dataset, which is community based model. Registering with this service, we will be able to get experimental and worked dataset[14].

Natural Language Toolkit, Scipy and other preprocessing packages are available. We can remove unstructured data from the dataset. Word count and token created for data set with the help of tokenizer.

As you can see in the fig.10 and 11, for training and testing 5 different classifiers are taken for the experiments. The output of this training and testing classified in 6 different human emotions. After probability distribution accuracy of these classifiers drawn[9][13].

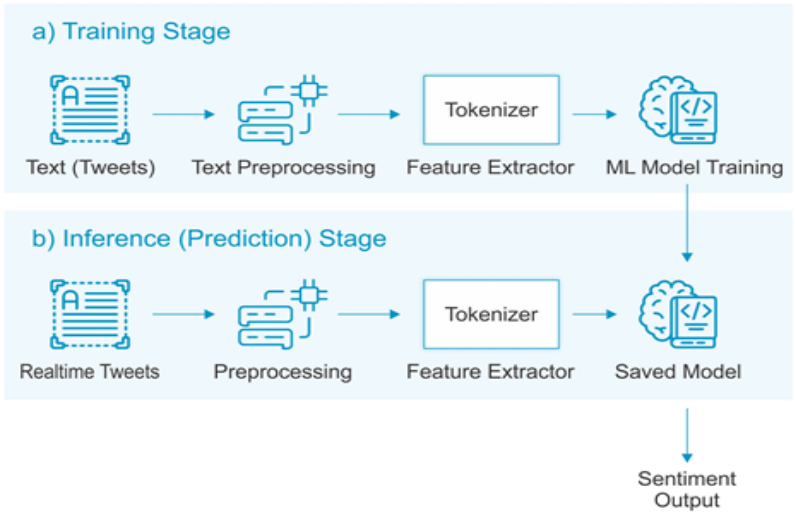


Fig. 11 Training and Prediction Phase of Covid-19 Twitter Data Sentiment Analysis

IV. RESULTS

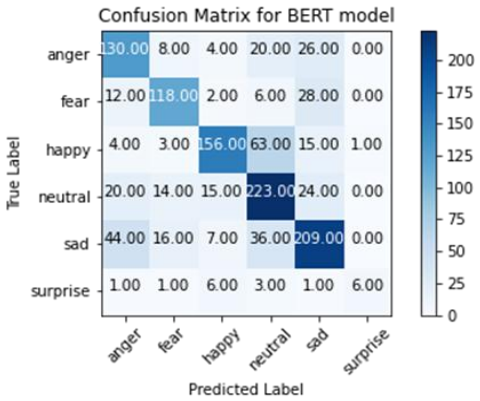


Fig. 12 BERT model Confusion matrix

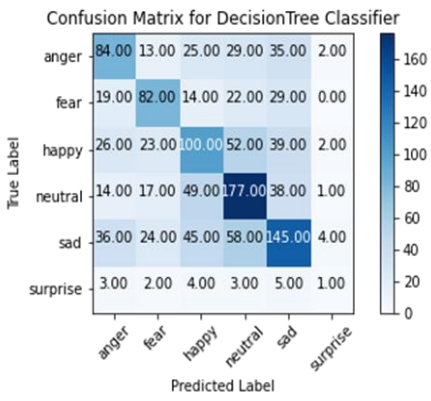
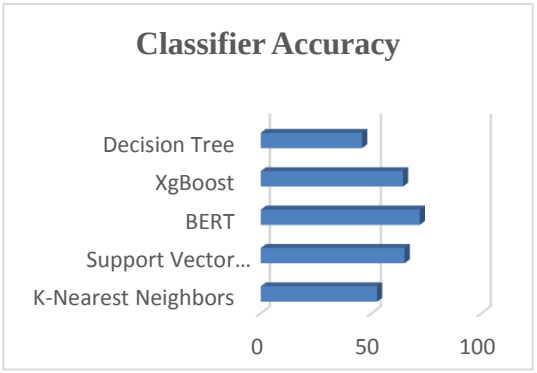


Fig. 13 DT Confusion Matrix



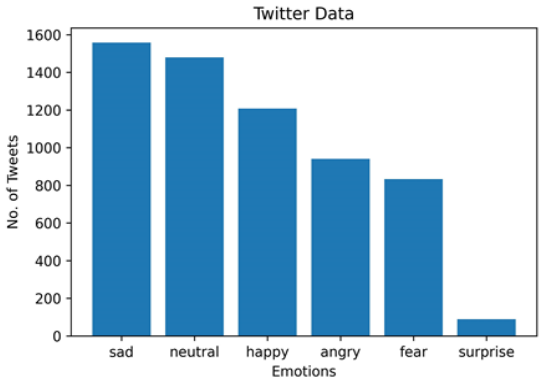


Fig. 14 Five Classifier Accuracy

Fig. 15 Human Emotion based on experiment

As we can see in fig. 12 and 13 we have carried out confusion matrix for 5 classifiers. After Calculating precision, Recall, F-1 score for 5 classifiers, we came to know that BERT classifier gives best result in terms of accuracy (Fig. 14). Result also shows people are sadder in Covid-19.

CONCLUSION

After Covid-19, it is very important to know human emotions based on twitter data. This work provides best result on five classifiers and on Kaggle dataset. In future, more classifiers will be experimented on same data.

REFERENCES

[1] Chau, Cruz, and Almazan, "Sentiment Analysis of Twitter Data Through Machine Learning Techniques," Software Engineering in the Era of Cloud Computing, pp. 185–209, 2020. Publisher: Springer, Cham.

[2] Kalai vani and Dinesh, "Machine Learning Approach to Analyze Classification Result for Twitter Sentiment," in 2020 International Conference on Smart Electronics and Communication (ICOSEC), (Trichy, India), pp. 107–112, IEEE, Sept. 2020.

[3] Wazery, Mohammed, Houssein, "Twitter Sentiment Analysis using Deep Neural Network," in 2018 14th International Computer Engineering Conference (ICENCO), (Cairo, Egypt), pp. 177–182, IEEE, Dec. 2018.

[4] Ranganathan and Tzacheva, "Emotion Mining in Social Media Data," Procedia Computer Science, vol. 159, pp. 58–66, Jan. 2019.

[5] JasiyaRaisa, Ulfat, Abdullah, Reza, "A Review on Twitter Sentiment Analysis Approaches", International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD) | 978-1-6654-1460-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICICT4SD50815.2021.9396915

[6] Roy, Ojha, "Twitter sentiment analysis using deep learning models", IEEE (INDICON) | 978-1-7281-6916-3/20/$31.00 ©2020 IEEE | DOI: 10.1109/INDICON49873.2020.9342279

[7] Nair, Veena, Aadithya, "Comparative study of Twitter Sentiment On COVID - 19 Tweets", (ICCMC) | 978-1-6654-0360-3/20/$31.00 ©2021 IEEE| DOI:10.1109/ICCMC51019.2021.9418320

[8] Elbagir and Yang, "Twitter Sentiment Analysis Using Natural Language Toolkit and VADER Sentiment", IMECS 2019, Hong Kong, March 13-15, 2019.

[9] Kavitha, Rajendran, and Varsha, "A correlation based SVM-recursive multiple feature elimination classifier for breast cancer disease using microarray."2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, 2016.

[10] Mao, Liu, "A BERT-based Approach for Automatic Humor Detection and Scoring," Proceedings of t helberian Languages Evaluation Forum (IberLEF 2019).

[11] Campos, Sastre, Francesc and Maurici and Bellver, Nieto, Xavier and Torres, Jordi, "Distributed training strategies for a computer vision deep learning algorithm on a distributed GPU cluster", journal Procedia Computer Science, volume 108, pages 315324, year 2017, publisher Elsevier.

[12] Jalil, Abbasi, Javed AR, Khan M, Abul Hasanat MH, Malik KM and Saudagar AKJ (2022), "COVID-19 Related Sentiment Analysis Using State-of-the-Art Machine Learning and Deep Learning Techniques." Front. Public Health 9:812735. doi: 10.3389/fpubh.2021.812735https://xgboost.readthedocs.io/en/stable/

[13] Kariya, "Twitter Sentiment Analysis", 2020 International Conference for Emerging Technology (INCET) Belgaum, India. Jun 5-7, 2020

[14] AlBadani, Dong J., "A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM." Appl. Syst. Innov. 2022, 5, 13. https://doi.org/10.3390/asi5010013

[15] Hase Sudeep Kisan, Hase Anand Kisan, Aher Priyanka Suresh, "Collective intelligence & sentiment analysis of twitter data by using StandfordNLP libraries with software as a service (SaaS)", 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), DOI: 10.1109/ICCIC.2016.7919697, Electronic ISSN: 2473-943X