ISSN 2007-9737

Multi-Class Sentiment Analysis of COVID-19 Tweets by Machine Learning and Deep Learning Approaches

Maaskri Moustafa[1,*], Sid Ahmed Mokhtar-Mostefaoui[1], Madani Hadj-Meghazi[1], Mohamed Goismi[2]

[1] University of Tiaret, LRIAS Laboratory, Computer Science Department, Algeria

[2] Dr.Tahar Moulay University GeCoDe Laboratory, Computer Science Department, Algeria

{moustafa.maaskri, h.meghazi}@univ-tiaret.dz, s_mostefaoui@esi.dz, mohamed.goismi@univ-sba.dz

Abstract. COVID-19 is a virus that has spread rapidly over the globe. The condition has repercussions beyond the realm of public health. Twitter is one platform where people post reactions to events during the outbreak. User-generated information, like tweets, presents unique challenges for sentiment analysis on Twitter data. With that in mind, this work employs four methods for analyzing Twitter data in terms of sentiment: the vector space model (TF-IDF) with three different ensemble machine learning models (voting, bagging, and stacking) and BERT (Bidirectional Encoder Representations from Transformers). Experiments showed that BERT outperformed the other three techniques, with an F1-score of 74%, a precision of 74%, and a recall of 74% for categorizing five sentiment classes on data from a Kaggle competition (Coronavirus tweets NLP-Text Classification).

Keywords. Ensemble machine learning, deep learning, voting, bagging, stacking, BERT.

1 Introduction

Several social media platforms are generating enormous volumes of text data these days, which has sparked a renewed interest in data processing to uncover the data's underlying meaning in a broader setting. Because Twitter data are accessible to the public and handled transparently, they may be used to investigate novel natural language processing (NLP) and data mining approaches, such as sentiment analysis [4]. The personal information, opinion, or polarity communicated in phrases or paragraphs may be extracted via sentiment analysis.

A valuable technique that offers real-time monitoring and decision-making capacities in the battle against the COVID-19 epidemic is sentiment analysis of data from social media platforms such as Twitter. This kind of analysis may be used to extract information from raw data. Many nations have implemented steps like isolation, quarantine, lockdown, or social distancing to address social media fears about the COVID-19 pandemic [13, 18].

However, different ethnicities and cultures have different methods of expressing their ideas. No matter the topic (health, politics, sports, or entertainment), people in one nation may react more passionately than others. Data-driven machine learning (ML) techniques predict [11, 23].

ML algorithms are widely utilized in health informatics [12, 5], pandemic predictions [13, 31], autism prediction [16], and many other fields. Many researchers have used ML systems to analyze Twitter sentiment. Villavicencio et al. [29] used

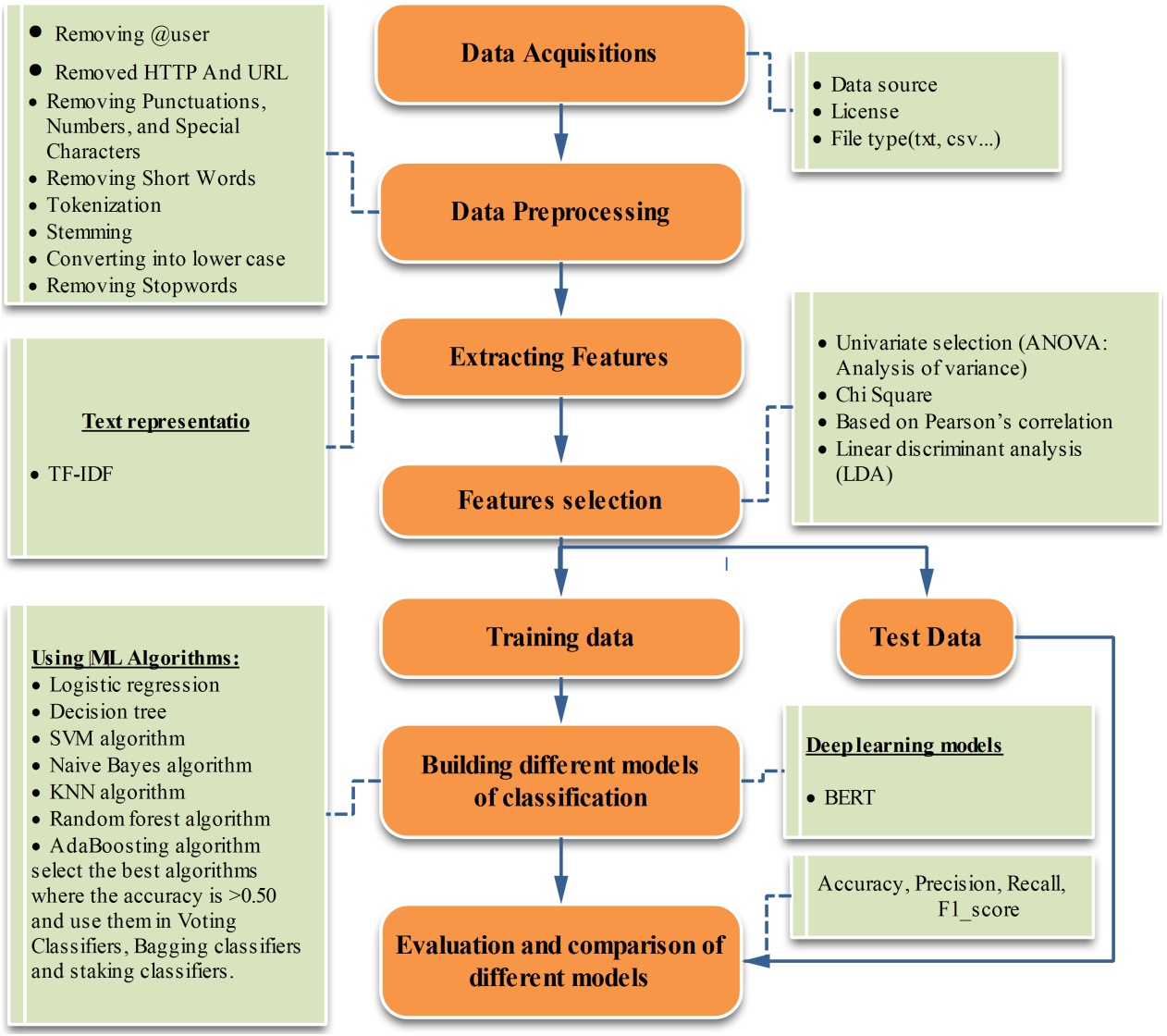

Fig. 1. Proposed scheme diagram

the Naïve Bayes classifier to analyze COVID-19 vaccination tweets in the Philippines and obtained 81.77% accuracy. The classifier was tested on 11,974 manually tagged tweets. Khan et al. [17] used the Naïve Bayes classifier to sentiment score 50,000 COVID-19 tweets and discovered 19% positive and 70% negative tweets. The authors of [15] employed deep learning classifiers to categorize 600 COVID-19-related tweets by sentiment. H-SVM had the most remarkable accuracy (86%), recall (69%), and F1-score (77%), among the classification methods employed in their research. Gupta et al. [9] investigated Twitter users' perceptions of the impact of weather on SARS-CoV-2 transmission.

The research filtered relevant tweets (n = 28555) using 11 ML algorithms and classified annotated tweets (n = 2442) into sentiment labels. The relevant tweet dataset showed 40.4% ambiguity regarding weather's influence, 33.5%

Table 1. Sample Tweet data for sentiment classification

OriginalTweet	Sentiment
The Home Depot is limiting the number of customers allowedinto its stores at any one time	Positive
I SERIOUSLY DOUBT anyone will be voting for ANY RepublicanPlease wear a mask take hand sanitizer and vote these bastards out	ExtremelyNegative
I thought I would save more money by being quarantined butonline shopping determined that was a lie. ???\r\r\n #CoronaCrisis	Extremely Positive

no effect, and 26.1% some effect on SARS-CoV-2 transmission. Latent Dirichlet Allocation (LDA) modeling was used to identify COVID-19-related topics from Twitter data [6, 1].

The researchers in [27] assessed machine learning classifiers on 7528 COVID-19 tweets. Automatic Twitter annotation yielded 93% accuracy in the trial. This research indicated that ML techniques were widely employed for COVID-19 tweet sentiment analysis and categorization. Due to the COVID-19 epidemic, no research has explicitly investigated ensemble ML models for sentiment analysis.

Nemez [22] employed a trained Recurrent Neural Network (RNN) to assess the percentage of positive, neutral, and negative attitudes in a coronavirus-related Twitter dataset. RNN forecasts showed 24.8% more positive tweets on May 13-14, 2020. Rustam [27] examined RF, XGBoost, SVC, ETC, DT, and LSTM for sentiment analysis. LSTM performed worse in that trial. The training data for the LSTM were insufficient.

Chakraborty [3] used a fuzzy approach using a Gaussian membership function to predict Twitter sentiment with 79% accuracy. According to particular research, sentiment analysis on Twitter data is difficult owing to the diversity, writing faults, and non-standard sentence patterns of user-generated information. This study analyzes COVID-19-related Twitter data using ensemble machine-learning methods and deep-learning models.

Voting, bagging, stacking, and BERT (Bidirectional Encoder Representations from Transformers) were tested for COVID-19 Twitter sentiment analysis. Coronavirus tweets' NLP-Text Classification Kaggle competition data already contains a sentiment class [20].

The following section presents the proposed scheme for sentiment analysis of COVID-19 tweets. Furthermore, Section 3 discusses the ML and deep learning model findings in detail. The final section concludes the article and highlights its limitations.

1.1 Proposed Scheme

The methodological overview of the sentiment analysis process is shown in figure 1. In the data accession step, the COVID-19-related tweets data were collected from Twitter. Moreover, the collected dataset was preprocessed, followed by word representation, classification methods, and performance measurement.

1.2 Data Acquisition

We have collected English-language tweets related to the coronavirus that were posted on Twitter between January 1, 2020, and December 31, 2020, sourced from several countries around the world through the pandemic timeline, and they are available at [20].

A set of predefined and widely used science and news media terms related to coronavirus, such as "COVID-19", "coronavirus", "lockdown", "isolation", "quarantine", "pandemic" and "ncov-2020" was used to collect tweets.

The data consisted of training data (41157) and testing data (3798). The sample of the tweets and the sentiment classes, according to Table 1 shows the sample Tweet Data for Sentiment

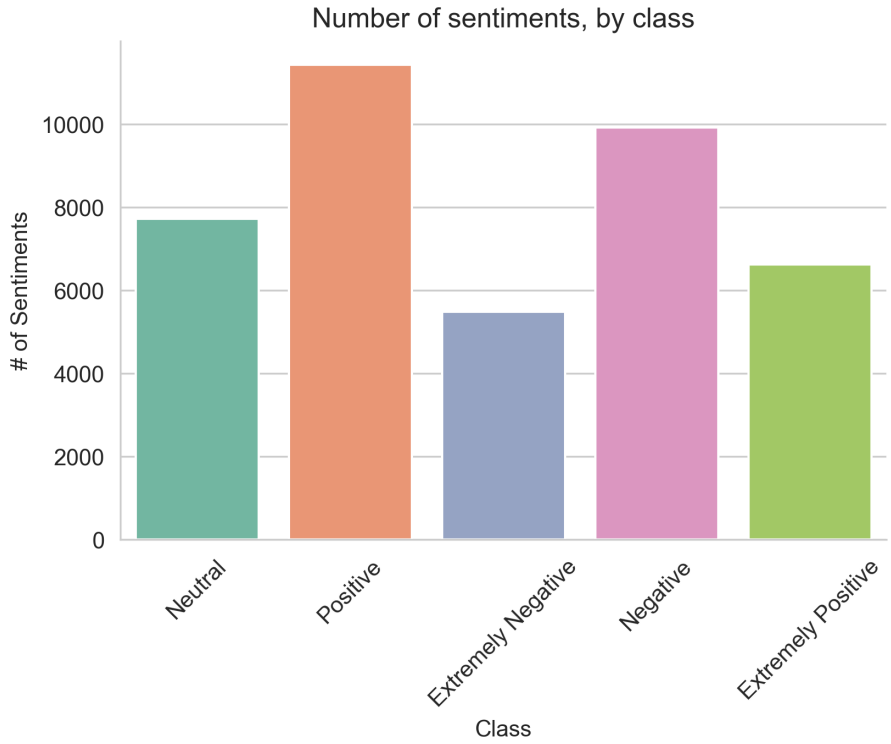

Fig. 2. Class distribution of the dataset

Classification, and Figure 2, shows the data distribution in each class for data with five classes.

1.3 Data Preprocessing

Raw data must be treated in a preprocessing stage before it can be successfully used with machine learning algorithms. This stage prepares the data to be used. This system performs its data preprocessing with the assistance of Natural Language Processing [24].

The text data are, first and foremost, changed to lowercase during this stage. This form has all stop words eliminated, and the corresponding contractions have been changed. In the Python NLTK package, a list of stop words is defined, which is used in this procedure.

Additionally, a custom function is developed to substitute contractions to finish the job. In order to reduce the likelihood of confusion, a check for spelling errors is carried out. The first step is to replace uppercase with lowercase. Following this step, the text will have any special characters, URLs, HTML tags, and stop words removed.

The text data is subjected to one more round of tokenization [14], normalization, and lemmatization. When it comes to natural language processing, there are three critical functions known as stemming, tokenization, and normalization used for preprocessing text before classification.

1. Tokenization: In natural language processing (NLP), tokenization divides text content into smaller components. A token is a name given to each unit. Every single word is turned into a token for this work [8].
2. Stemming: In stemming, the morphological forms of a word are converted back to their stems under the assumption that each form is semantically related to the others. The stem does not need to be a term already present in the dictionary. Nevertheless, after stemming is complete, all of the stem's variants should map

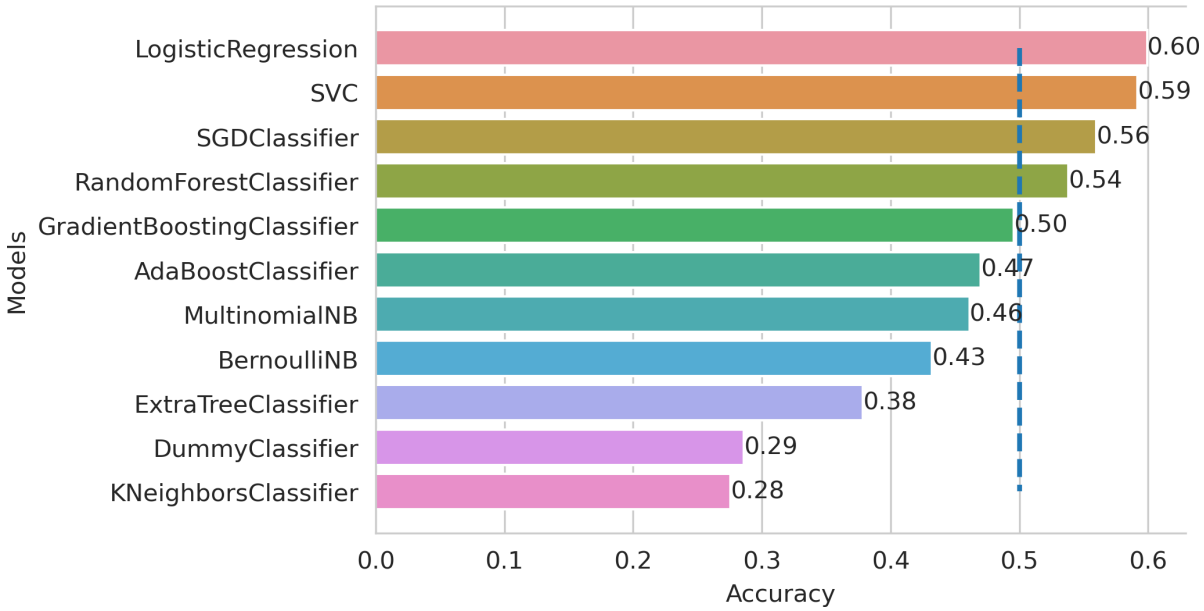

Fig. 3. The results obtained by the different machine learning models

to this form. When utilizing a stemmer, two things need to be taken into consideration [21]:

(a) It is reasonable to presume that the various morphological variants of a word have the same core meaning, and they should thus all be mapped to the same stem.

(b) It is essential not to confuse words that do not have the same meaning with one another.

These two rules are sufficient so long as the stems produced are helpful for the programs we use for text mining and language processing. In most contexts, stemming is understood to function as a mechanism that improves recall. Compared to languages with a more complex morphology, the influence of stemming is not as strong in languages with a relatively simple morphology.

3. Normalization: It is the process of converting an odd text into its typical form.

   People occasionally use a term unconventionally to convey their meaning [19]. This content has to be reformatted into its proper form, and any spelling errors need to be corrected.

4. Extracting Features: By extracting features from text and representing them as a vector of real numbers, a procedure known as “text feature extraction” can be performed [26, 10].

In this study, we used a technique called TF-IDF that generates a vector containing a set of real-valued features for each text, with the value of each feature depending on how often a specific word occurs in the text.

2 Building Models

Four different ML models were built using preprocessed tweets. The ML models were trained using the training dataset, while the performance of the models was evaluated using both the training and test datasets. The ML models are analyzed in detail in the following subsection.

Table 2. Performance measures for the proposed models

Models	Accuracy	Precision	Recall	F1-score
Voting Classifier	0.60	0.62	0.63	0.64
Bagging Classifier	0.61	0.61	0.63	0.65
Stacking Classifier	0.64	0.64	0.65	0.66
BERT (Base)	**0.73**	**0.74**	**0.74**	**0.74**

2.1 Analyzing Machine Learning Models

2.1.1 Voting

A voting ensemble technique is a machine learning model that produces a single final prediction by combining the predictions of multiple machine learning models [28]. Because all the training data were used to train the models with this ensemble method, they should each have their personality. When performing regression tasks, the result is the mean of the predictions made by the models.

Instead, two methods are available: hard voting and soft voting, which can be used to estimate the final output of classification problems. Voting's primary purpose is to enhance generality by correcting flaws specific to each model. This is especially important when the models perform well on a predictive modeling problem.

2.1.2 Bagging

Bagging subsamples of training data to improve one classifier's generalization performance. Overfitting models benefit from this strategy. Bagging data from subsamples includes bootstrapping and aggregating. This method uses random sampling with replacement to resample the data, which overlaps training data. Regression voting or classification voting yields the final prediction for each data set. This strategy improves very little because the classifier's hyperparameters do not vary from one subsample to another. This bias-reduction strategy is expensive and will not help with volatility. It reduces variance by better generalizing when the data is overfitted but not under fitted.

2.1.3 Stacking

Stacking ensemble models employ weighted voting to avoid all models contributing equally to the forecast. Stacking models have base models and meta-models (models that learn how to combine the predictions of the base models). Linear regression is used for regression, and logistic regression for classification. Out-of-sample base model predictions teach the meta-model. In other words, (1) data not used to train the base models are fed to them, (2) they make predictions, and (3) these predictions and the ground truth labels are utilized to fit the meta-model. Regression problems use predicted values. The affirmative class prediction is usually the input for binary classification problems. Finally, the multi-class classification uses the projected values for all classes.

2.1.4 BERT Classifier

BERT is a deep learning model that excels at NLP tasks. One output layer may fine-tune BERT's deep bidirectional representation [2]. This paper used BERT-Base. Moreover, BERT-Base has 12 layer/transformer blocks, 768 hidden units, and 12 self-attention heads with 110 M optimized parameters. BERT employs a 30000-word set of fundamental embeddings [30].

The input representation is the token, segment, and position embeddings total. Furthermore, for preprocessing data, both [CLS] and [SEP] were used as a classification token and a sentence marker, respectively. Additionally, the sentiment categorization output layer comprises [CLS] representation.

3 Experimental Results and Discussion

This section briefly discusses the study of different ML ensemble algorithms for the category of user sentiment under different labels (extremely positive, positive, extremely negative, negative, and neutral). The ML models were created and examined using the scikit-learn [25] package and the Python programming language.

The manually labeled dataset was split 80/20 randomly between the training and testing phases. As a result, 80% of the data were classified as training data, and 20% as testing data. The grid search tuning approach [7] was used to tune the hyperparameters, which can regulate how the algorithms learn, to identify the best hyperparameters for the utilized models.

The algorithms' performance was evaluated using precision, recall, and the F1-score. The experiment was conducted to discover the best parameters for each method used to classify the sentiment data of the COVID-19 tweets. Tweets with five classes were used in the experiment.

The first time we used the popular machine learning algorithms with the data representation of TF-IDF, the figure 3 shows the results obtained, such as the algorithms used: Logistic Regression (Lr), Support Vector Machine (SVM), Naïve Bayes (NB), K-Nearest Neighbors (KNN), Random Forest, Gradient Boosting, and AdaBoosting. Then the best models (such as those with accuracy above 50%) were selected to build the ensemble models (voting, bagging, and staking). Finally, the BERT models were used.

3.1 Voting Classifier (VC) Setup

To obtain the final predicted labels, hard voting, also known as majority voting, was used in this study among the Decision Tree (DT), Support Vector Classifier (SVC), and Logistic Regression (LR). The precision, recall, and F1-score for the VC model on the test dataset were 98.9%, 99.5%, and 99.3%, respectively.

3.2 Bagging Classifier (BC) Setup

The outputs from the predictive models are then applied to a voting scheme for better categorization. The basic estimator for training the BC model in this investigation was a Logistic Regression with $n_{\text{estimators}} = 100$. The bagging classifier's accuracy, precision, recall, and F1-score on the testing dataset were 61%, 61%, 63%, and 61%, respectively (see Table 2).

3.3 Stacking Classifier (SC) Setup

The proposed SC model's design consisted of two levels. The VC and BC models discussed above made up the first layer of the SC model, and a logistic regression model made up the second layer.

For every observation and test in the dataset, two distinct models were used to generate the conclusions. The judgments attained by these methods served as input features for the second-layer LR model.

The second-layer model then delivered the result based on the input features. The SC model's accuracy, precision, recall, and F1-score were 64%, 64%, 65%, and 65% on the training set, respectively (see Table 2).

3.4 BERT Setup

The BERT process was divided into two stages: pre-training and fine-tuning. The BERT architecture was trained on several tasks using unlabeled data during the pre-training phase. This was achieved so that it could be used later. Then, for fine-tuning, BERT was trained on the data utilized in this research, namely the tweets from the COVID-19 event. During the fine-tuning process, the used parameters included learning rates of 10-5, a batch size of 32, and a maximum iteration of 15 epochs.

Within the framework of the sentiment categorization method that uses BERT, the following parameters were observed: (a) the total number of effective classes is five; (b) the learning rate is 10e-5. Table 2 and figure 4 present the findings of the performance evaluation of the sentiment categorization using BERT.

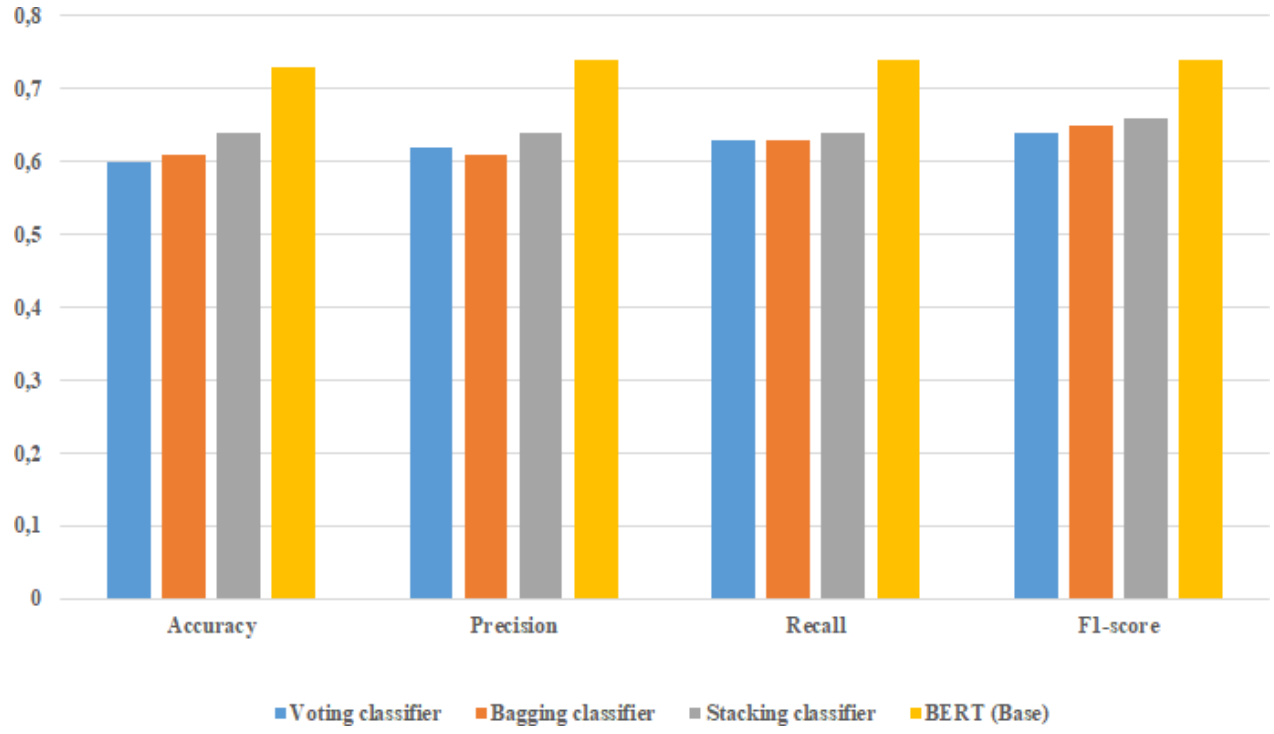

Fig. 4. Performance of different models

The best performance achieved was 0.74 (precision), 0.74 (recall), and 0.74 (F1-score) for classifying the five sentiment classes.

4 Conclusion

In this paper, the sentiment classification of the COVID-19 tweets dataset was investigated by comparing two sentiment classification schemes. The first scheme included ensemble ML models to classify tweets into five classes.

The Stacking Classifier showed the highest F1 score of 65% in this scheme, while Voting Classifier and Bagging Classifier models showed promising results, indicating that ensemble ML models can be used for sentiment analysis. The second scheme is sentiment classification using BERT.

The classification results achieved by BERT were better than the first scheme, reaching 74% (F1-score), 74% (precision), and 74% (recall) for the classification of five sentiment classes. Future studies may focus on trying different encoders, such as the variants of BERT and Word2vec, for text embedding to find the best suitable encoding for the classifiers and get better outcomes.

References

1. **Abd-Alrazaq, A., Alhuwail, D., Househ, M., Hamdi, M., Shah, Z. (2020).** Top concerns of tweeters during the COVID-19 pandemic: Infoveillance study. Journal of Medical Internet Research, Vol. 22, No. 4, pp. e19016. DOI: 10.2196/19016.
2. **Bozuyla, M., Ozundefinedift, A. (2022).** Developing a fake news identification model with advanced deep languagetransformers for turkish covid-19 misinformation data. Turkish Journal of Electrical Engineering and Computer Sciences, Vol. 30, No. 3, pp. 908–926. DOI: 10.55730/1300-0632.3818.
3. **Chakraborty, K., Bhatia, S., Bhattacharyya, S., Platos, J., Bag, R., Hassanien, A. E.**

(2020). Sentiment analysis of covid-19 tweets by deep learning classifiers—a study to show how popularity is affecting accuracy in social media. Applied Soft Computing, Vol. 97, pp. 106754. DOI: 10.1016/j.asoc.2020.106754.

4. **Chong, W. Y., Selvaretnam, B., Soon, L. K. (2014).** Natural language processing for sentiment analysis: An exploratory analysis on tweets. 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology, pp. 212–217. DOI: 10.1109/icaiet.2014.43.
5. **Dhaya, R. (2020).** Deep net model for detection of COVID-19 using radiographs based on ROC analysis. Journal of Innovative Image Processing, Vol. 2, No. 3, pp. 135–140. DOI: 10.36548/jiip.2020.3.003.
6. **Garcia, K., Berton, L. (2021).** Topic detection and sentiment analysis in twitter content related to covid-19 from Brazil and the USA. Applied Soft Computing, Vol. 101, pp. 107057. DOI: 10.1016/j.asoc.2020.107057.
7. **Ghawi, R., Pfeffer, J. (2019).** Efficient hyperparameter tuning with grid search for text categorization using kNN approach with BM25 similarity. Open Computer Science, Vol. 9, No. 1, pp. 160–180. DOI: 10.1515/comp-2019-0011.
8. **Ghulam, H., Zeng, F., Li, W., Xiao, Y. (2019).** Deep learning-based sentiment analysis for roman urdu text. Vol. 147, pp. 131–135. DOI: 10.1016/j.procs.2019.01.202.
9. **Gupta, M., Bansal, A., Jain, B., Rochelle, J., Oak, A., Jalali, M. S. (2021).** Whether the weather will help us weather the covid-19 pandemic: Using machine learning to measure twitter users' perceptions. International Journal of Medical Informatics, Vol. 145. DOI: 10.1016/j.ijmedinf.2020.104340.
10. **Imran, M., Afzal, M. T., Qadir, M. A. (2017).** A comparison of feature extraction techniques for malware analysis. Turkish Journal of Electrical Engineering and Computer Sciences, Vol. 25, pp. 1173–1183. DOI: 10.3906/elk-1601-189.
11. **Islam, M. N., Inan, T. T., Rafi, S., Akter, S. S., Sarker, I. H., Islam, A. K. M. N. (2020).** A systematic review on the use of AI and ML for fighting the COVID-19 pandemic. IEEE Transactions on Artificial Intelligence, Vol. 1, No. 3, pp. 258–270. DOI: 10.1109/tai.2021.3062771.
12. **Islam, M. N., Mahmud, T., Khan, N. I., Mustafina, S. N., Najmul-Islam, A. K. M. (2021).** Exploring machine learning algorithms to find the best features for predicting modes of childbirth. IEEE Access, Vol. 9, pp. 1680–1692. DOI: 10.1109/ACCESS.2020.3045469.
13. **Islam, M. N., Najmul-Islam, A. K. M. (2020).** A systematic review of the digital interventions for fighting COVID-19: The Bangladesh perspective. IEEE Access, Vol. 8, pp. 114078–114087. DOI: 10.1109/access.2020.3002445.
14. **Javed-Mehedi-Shamrat, F. M., Tasnim, Z., Ghosh, P., Majumder, A., Hasan, M. Z. (2020).** Personalization of job circular announcement to applicants using decision tree classification algorithm. IEEE International Conference for Innovation in Technology, pp. 1–5. DOI: 10.1109/inocon50539.2020.9298253.
15. **Kaur, H., Ahsaan, S. U., Alankar, B., Chang, V. (2021).** A proposed sentiment analysis deep learning algorithm for analyzing covid-19 tweets. Information Systems Frontiers, Vol. 23, No. 6, pp. 1417–1429. DOI: 10.1007/s10796-021-10135-7.
16. **Kazi-Shahrukh, O., Prodipta, M., Nabila-Shahnaz, K., Md.-Rezaul, K. R., Md-Nazrul, I. (2019).** A machine learning approach to predict autism spectrum disorder. pp. 1–6. DOI: 10.1109/ECACE.2019.8679454.
17. **Khan, R., Shrivastava, P., Kapoor, A., Tiwari, A., Mittal, A. (2020).** Social media analysis with AI: Sentiment analysis

techniques for the analysis of twitter COVID-19 data. Journal of Critical Reviews, Vol. 7.

18. **Laato, S., Islam-Najmul, A. K. M., Islam, M. N., Whelan, E. (2020).** What drives unverified information sharing and cyberchondria during the COVID-19 pandemic?. European Journal of Information Systems, Vol. 29, No. 3, pp. 288–305. DOI: 10.1080/0960085x.2020.1770632.
19. **Mehta, R. P., Sanghvi, M. A., Shah, D. K., Singh, A. (2019).** Sentiment analysis of tweets using supervised learning algorithms. Springer Singapore, pp. 323–338. DOI: 10.1007/978-981-15-0029-9_26.
20. **Miglani, A. (2020).** Coronavirus tweets NLP - Text classification. www.kaggle.com/datasets/datatattle/covid-19-nlp-text-classification.
21. **Moral, C., de-Antonio, A., Imbert, R., Ramírez, J. (2014).** A survey of stemming algorithms in information retrieval. Information Research, Vol. 19, No. 1.
22. **Nemes, L., Kiss, A. (2020).** Social media sentiment analysis based on covid-19. Journal of Information and Telecommunication, Vol. 5, No. 1, pp. 1–15. DOI: 10.1080/24751839.2020.1790793.
23. **Nichols, J. A., Herbert-Chan, H. W., Baker, M. A. B. (2019).** Machine learning: applications of artificial intelligence to imaging and diagnosis. DOI: 10.1007/s12551-018-0449-9.
24. **Pang, B., Lee, L. (2004).** A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. pp. 271–278. DOI: 10.3115/1218955.1218990.
25. **Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Müller, A., Nothman, J., Louppe, G., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E. (2012).** Scikit-learn: Machine learning in python. Journal of Machine Learning Research, Vol. 12, pp. 2825–2830.
26. **Perkins, J. (2014).** Python 3 text processing with NLTK 3 cookbook. Packt Pub Ltd.
27. **Rustam, F., Khalid, M., Aslam, W., Rupapara, V., Mehmood, A., Choi, G. S. (2021).** A performance comparison of supervised machine learning models for covid-19 tweets sentiment analysis. PLOS ONE, Vol. 16, No. 2, pp. e0245909. DOI: 10.1371/journal.pone.0245909.
28. **Ruta, D., Gabrys, B. (2005).** Classifier selection for majority voting. Information Fusion, Vol. 6, No. 1, pp. 63–81. DOI: 10.1016/j.inffus.2004.04.008.
29. **Villavicencio, C., Macrohon, J. J., Inbaraj, X. A., Jeng, J. H., Hsieh, J. G. (2021).** Twitter sentiment analysis towards COVID-19 vaccines in the Philippines using Naïve Bayes. Information, Vol. 12, No. 5, pp. 204. DOI: 10.3390/info12050204.
30. **Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, L., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., et al. (2016).** Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv, pp. 1–23. DOI: 10.48550/arXiv.1609.08144.
31. **Zaman, A., Muhammad-Nazrul, I., Zaki, T., Sajjad-Hossain, M. (2020).** ICT intervention in the containment of the pandemic spread of COVID-19: An exploratory study. arXiv, pp. 1–16. DOI: 10.48550/arXiv.2004.09888.

Article received on 08/04/2023; accepted on 18/04/2024.
** Corresponding author is Maaskri Moustafa.*