Sentiment Analysis of Tweets Using Deep Learning

Jaishree Ranganathan(✉) and Tsega Tsahai

Middle Tennessee State University, Murfreesboro, TN 37132, USA
Jaishree.Ranganathan@mtsu.edu

Abstract. The coronavirus pandemic has caused a worldwide crisis and a drastic change in day-to-day life activities. Worldwide, people use social media platforms to share and discuss their opinions about the situation. Twitter is one such platform for public conversation around the coronavirus pandemic, the spread of disease, vaccination, non-pharmaceutical interventions, and many other discussions. In this study, we use Twitter social medial data for sentiment analysis. The tweets are collected based on covid-19 related hashtags. This work presents a deep learning-based framework for sentiment analysis using DistilBERT, a distilled version of Bidirectional Encoder Representation from Transformers (BERT), Convolutional Neural Network (CNN), and Long Short Term Memory (LSTM). The results show that transformer-based pre-processing and fine-tuning yield better performance results. The DistilBERT model yields the highest accuracy of 91.46% compared to the CNN and LSTM models.

Keywords: BERT · Coronavirus · Convolutional Neural Network · Long Short Term Memory · Tweets · Sentiment analysis

1 Introduction

Covid-19 disease outbreak has caused changes in several aspects of people's lives worldwide. There are protocols in place for medical and public health systems due to the risks posed by the coronavirus disease outbreak. Historical data and computer technologies have been helpful in the decision-making process during such infectious disease outbreaks in the past [2]. The emergence of web 2.0 has led to more user-generated content and usability for end-users compared to the earlier web 1.0. Some of the popular platforms of user-generated content on the web include blogs, forums, and social networking sites like Twitter and Facebook [1]. Thus there is an increase in the number of social media platforms and the users of these forums worldwide. A tremendous amount of user-generated content is available on Twitter, Facebook, and other venues like e-commerce websites and news portals. Such information is considered an asset of data by businesses, individuals, and other entities looking for timely feedback. These resources are extantly used to understand general public opinions. Some of the most common applications of sentiment analysis include consumer product reviews, customer service, and stock markets [9]. It is evident that such social media platforms have

W. Chen et al. (Eds.): ADMA 2022, LNAI 13725, pp. 106–117, 2022.
https://doi.org/10.1007/978-3-031-22064-7_9

also become significant sources in reflecting real-world events [5]. For instance, social media has become a blazing platform with many discussions and opinions related to coronavirus worldwide.

Twitter is one of the popular microblogging platforms in everyday life for people regardless of their geographic location [21]. Twitter data has played a significant role in the past for several epidemics, outbreak predictions, and monitoring public health [29]. Content on such web platforms includes a variety of raw, unstructured data formats, including text, images, video, and audio. Advancement in computer technologies and techniques, including artificial intelligence and machine learning, provides the power to process such unstructured data and gain valuable insights [8]. Natural language processing (NLP) is a subfield of artificial intelligence constantly gaining attention in almost every domain, including public health, business, and education. Sentiment analysis is one of the trending and most studied research areas in natural language processing and machine learning. In this study, we use tweets collected using hashtags related to coronavirus. We present a deep learning-based framework for sentiment analysis of tweets. We also present a comparative study of results obtained using several deep learning models. The models include convolutional neural network (CNN), long short-term memory (LSTM), and distiled version of bidirectional encoder representation from transformer (BERT). The rest of the paper is organized as follows; Sect. 2 describes the related work, Sect. 3 - describes data collection and pre-processing, Sect. 4 - describes methodology, Sect. 5 - describes experiments and results, and Sect. 6 - conclusions.

2 Related Work

2.1 Sentiment Analysis on Tweets

Sentiment analysis of text, especially tweets, is a trending area of research with a range of machine learning models for automated text sentiment classification. However, it is important to have a labeled dataset for supervised classification. Several works in the literature use emoticons [11], dictionary or lexicons [23] as source of labeling data. Then use traditional machine learning classifiers like Naive Bayes, Maximum Entropy, SVM, Decision Tree, Random Forest, and Decision Table Majority.

Similarly, most of the work applying deep learning models like neural networks uses pre-existing annotated datasets for training and testing purposes. For instance, [16] use standard Twitter sentiment datasets like Stanford Twitter Sentiment Test (STSTd), SE2014 from SemEval2014, Stanford Twitter Sentiment Gold (STSGd), Sentiment Evaluation Dataset (SED), Sentiment Strength Twitter Dataset (SSTd). They achieved the highest accuracy of 87.62% using Glove embedding and Deep Convolutional Neural Network (CNN) on the STSTd dataset. Also in [7], the authors use Stanford Sentiment Treebank (SSTb) movie reviews and Stanford Twitter Sentiment corpus (STS) Twitter messages. They present Character to sentence CNN for sentiment prediction and achieve the highest accuracy of 86.4% on the Twitter sentiment corpus. Another study [17], uses STS Gold Dataset and Movie Review data for training the CNN model and achieve an accuracy of 75.39%.

Some of the works use SemEval datasets. For example, [12] utilize SemEval2014 Task9-SubTask B full data, the SemEval2016 full data Task4 and the SemEval2017 development data for the model train and test. They use Glove and word2vec embedding for CNN and LSTM and achieve the highest accuracy of 59% using multiple CNN and bi-LSTM networks; and [31] use benchmark sets from the SemEval 2015 dataset, which are manually annotated into positive, negative, and neutral labels. They show that deep convolutional neural networks with pre-trained vectors like Glove achieve an F1 score of 64.85%. Similarly, research indicates the use of other pre-annotated datasets for the sentiment classification of tweets. For instance, [26], use the train and dev corpora from Twitter'13 to 16 for training and Twitter'16-dev as a development set. They use Lexical, part-of-speech, and sentiment embeddings to train the CNN model with the fusion input and achieve an F-1 score of 63%. Authors in [34] present a word-character-based CNN and utilize pre-annotated Twitter corpus. They achieve the highest accuracy of 0.8119 for polish language sentiment classification.

Recently transformer-based models for sentiment classification on tweets have been explored. For instance, [22] use the BERT base model for the classification of positive and negative sentiments of Italian tweets based on the SENTIPOLC 2016 corpus. They achieve an F-score of 0.75. In [10], the authors propose a target-dependent BERT model for sentiment classification and achieve an accuracy of 77.31% on the pre-existing twitter dataset.

2.2 Sentiment Analysis on Coronavirus Related Tweets

Covid-19 caused an unprecedented impact worldwide, leading to public health emergencies, lockdowns, and other protective protocols. It is important to understand public opinions and concerns in such scenarios. Social media plays a critical role during such disease outbreaks. This section describes the existing literature that uses tweets related to coronavirus disease to perform sentiment analysis.

Several studies collect Twitter data using covid-related keywords. However, the need for annotated data for supervised classification is a limitation. Most studies use pre-existing tools or lexicons to label the collected tweets for further classification using a machine learning model. For instance, [4] use TextBlob and Afinn to label the tweets for supervised classification into positive, negative, and neutral categories. They propose a fuzzy rule-based model with 79% F1-score and compare bag-of-words and Doc2Vec using several classifiers, including Naive Bayes, Support Vector Machine (SVM), Ensemble models, multinominal, Bernoulli, and logistic regression classifiers. Similarly, [19] scrape tweets based on the keyword "coronavirus" and use VADER to label the tweets into positive, negative, and neutral. They use Long Short Term Memory (LSTM) and Artificial Neural Network (ANN). The neural network models achieve an accuracy of 84.5% and 76%, respectively. Authors in [25] crawl twitter data with keyword "COVID-19" using rapid miner tools and use Naive Bayes classification; Another study [13], extract tweets using keywords related to lockdown and annotate the data using TextBlob and Vader lexicons. They use eight different classifiers and achieve 84.4% as the best accuracy using LinearSVC classifier with unigram features; Similarly in [28], the authors use social distancing keywords to extract tweets and use SentiStrength to annotate the data. They further utilized SVM for sentiment classification with an accuracy of 71% for positive, negative, and neutral sentiments and 81% with only positive and negative tweets; Authors Villavicencio et al. [32] manually annotate tweets and employ Naive Bayes classification on English and Filipino tweets. Their model yield 81.77% accuracy.

Authors Sitaula and Shahi [30] use publicly available Nepali tweets dataset for sentiment classification. They use feature extraction with a multi-channel convolutional neural network on Nepali covid-19 related tweets, where they achieve an accuracy of 71.3%. Similarly, [15] train the deep learning model using the annotated Sentiment140 dataset. And then test the model with a new set of data related to covid tweets collected by the authors.

Research using Twitter data is most commonly used for real-time trend analysis. So, it is necessary to collect tweets based on specific keywords related to the event in consideration. Annotated data may not always be available in a similar domain. In such scenarios, the model may not be able to generalize for current new data, especially Twitter being the most active platform with millions of users. Also, existing works use word embeddings for deep learning-based classification. We collect Twitter data related to coronavirus disease using related hashtags. In this work we apply pre-processing for tweets explained in Sect. 3 and use the transformer based tokenizer (Sect. 4.1). We annotate data using RoBerTa base Twitter sentiment model [18]. Finally we present a deep learning-based framework for sentiment classification of text data.

3 Data Collection and Pre-Processing

Data Collection: The dataset for the experiments are tweets collected using the Tweepy library. We collect tweets using hashtags related to coronavirus between August 31, 2021 to November 26, 2021. The hahstags for the data collection is listed in the Table 1. Initial raw data collection was part of [24].

Table 1. Hashtags used in Data Collection.

Hashtags
#coronavirus, #covid, #COVID19, #corona,#pandemic, #coronaviruspandemic
#staysafe, #washyourhands, #disinfectant, #handwashing, #mask,
#ppe, #covidvaccine, #covidvaccination, #vaccine, #coronavirusvaccine,
#boostershots, #boostershot, #pfizerbooster, #deltavariant, #coviddeltavariant,
#SARSCoV2, #muvariant, #mu, #gammavariant, #gamma, #delta,
#stayhome, #quarantine, #lockdown, #stayathome, #socialdistancing

Pre-processing: Pre-processing is an essential step in natural language processing applications. It is the process of converting the raw form of text into a form suitable for specific tasks and retaining the original information.

The most common form of text pre-processing is lowercase every single token of the input text. Here the tweet text is lowercased to avoid sparsity in the dataset. Following that, we utilize Python's Regular Expression library to locate the URLs, and user mentions in the tweets and replace them with the words url and user respectively. Hashtags are essential in tweets, as users express their opinions or feelings through hashtags. So, we tokenize and replace the hashtags by using Python's wordninja library. For example, the hashtag #stayhome will be converted to (stay home).

Twitter users also express their emotions towards a topic using emoticons (emojis). Since we build a sentiment analysis model, we consider the emotion expressed through emoticons. However, we cannot use emoticons as input to text classification models. We use Regular Expression and Python's demoji library to convert the emoticons into phrases. Finally, to reduce the noise of the tweets, stop words were removed by using Pythons Natural Language Toolkit (NLTK). We remove the stop words as they add little to no value to a text. Figure 1 shows the model pipeline in this work.

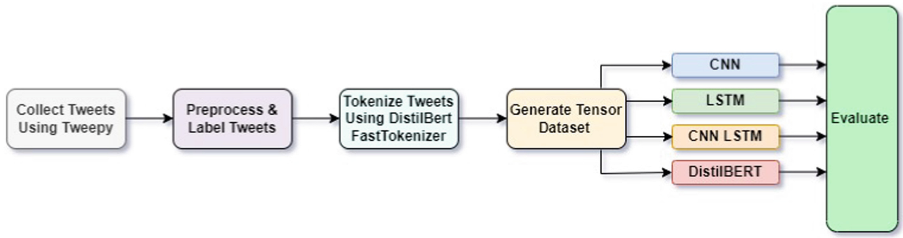

Fig. 1. Model framework.

Data Labeling: We present a supervised learning model framework in this work. We need labeled data for the classification model training and evaluation. We utilize the TextClassificationPipeline from Huggingface [33] to create a labeled dataset using the model [18]. The dataset now contains three class labels: positive, negative, and neutral.

Additional Pre-processing: Tweets are limited in the number of characters. In this process, we remove tweets with less than five words after initial pre-processing. In order to use a balanced dataset for the learning models, we perform random undersampling of the majority class (neutral and negative). Finally, we have 487,998 tweets in the dataset for use in the experiments. The sentiment distribution of this dataset is shown in Table 2. Then we shuffle the dataset to maintain the distribution of positive, negative, and neutral tweets across the train and test sets for further experiments.

Table 2. Sentiment distribution of tweets in the final dataset.

Class label	Number of instances
Positive	161,998
Negative	163,000
Neutral	163,000

4 Methodology

This paper presents a deep learning-based framework for classifying tweets into positive, neutral, and negative classes. Section 2.2 we see that most of existing studies on coronavirus related tweets are based on traditional machine learning models. Based on Sect. 2.1, where many deep learning models have proven to provide better results, in this study we experiment using Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Distiled version of Bidirectional Encoder Representation Transformer (DistilBERT) models. This section describes further text processing and the model architectures.

4.1 Text Tokenization and Padding

It is imperative that for the learning models to deal with text data, it needs some form of tokenization to convert the text to numerical representation. We tokenize the tweet text with FastTokenizer from the uncased DistilBERT model [27]. The tokenizer vocabulary contains 30,522 words. This tokenizer uses the concept of word piece tokenizer. The tweet text is of varying lengths. We use 100 as the maximum sequence length. The tokenized tweets are then truncated or padded for uniformity. These tokens are then mapped to their corresponding numerical IDs. Following that, the labels are one-hot encoded. The encoded label with corresponding tokenized tweets are converted into input pipelines by utilizing tensor flow tf.data.Dataset [20] API. This API creates an iterable dataset that is input to our text classification models.

4.2 Convolutional Neural Network Model (CNN)

Convolutional Neural Networks (CNN) is a type of feed-forward neural network that makes use of several hidden layers [3]. The hidden layers are typically the convolution, pooling, activation, dropout, and dense layers.

As shown in Fig. 2, the model starts with an input layer, followed by an embedding layer, three convolution layers with kernel sizes 3, 4, and 5, respectively, 32 filters, and relu as the activation function. These layers are followed

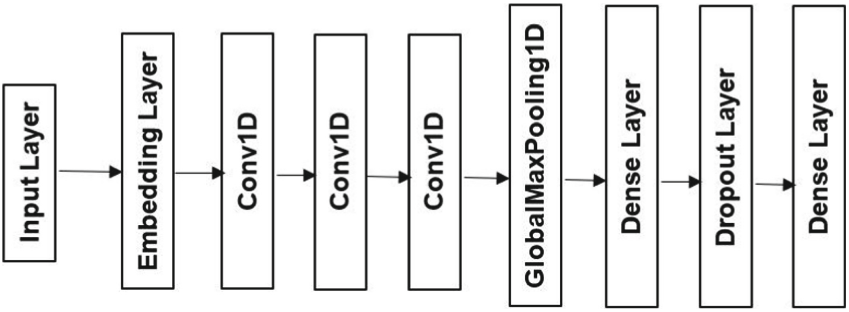

Fig. 2. CNN architecture.

by a 1D GlobalMaxPool layer, a Dense layer with 128 filters and relu activation function, a Dropout layer, and a final Dense layer with softmax activation to perform the classification into three classes. This model is compiled with the catagorical_crossentropy loss function and the Adam optimizer with a learning rate of 0.00001.

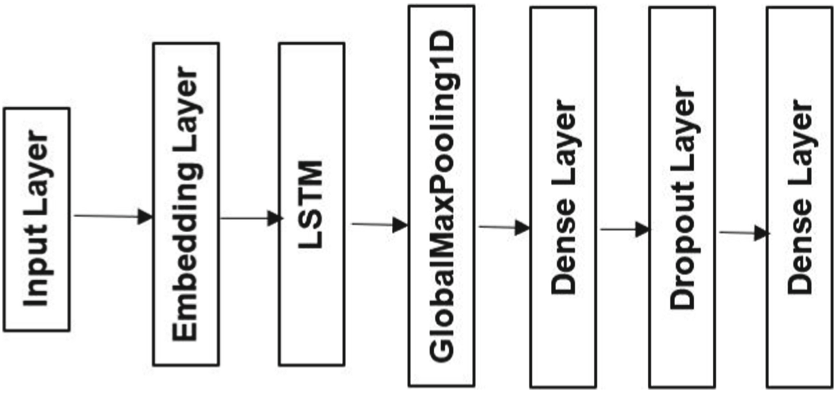

Fig. 3. LSTM architecture.

4.3 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) [14] is a type of Recurrent Neural Network (RNN) that is sequential and uses long memory for activation functions in hidden layers. LSTM has the ability to handle the vanishing gradient problem present in RNN.

The structure (See Fig. 3) of the LSTM includes an input layer, an embedding layer, followed by an LSTM layer. The output of this LSTM layer is then pooled using the 1D global max-pooling layer and passed to a dense layer with 64 units and the relu activation function and a dropout layer. Finally, a dense layer with softmax activation function. Similar to CNN model, the LSTM model is also compiled with the catagorical_crossentropy loss function and the Adam optimizer with a learning rate of 0.00001.

4.4 CNN-LSTM

This model utilizes a 3-layer 1-dimensional CNN with three different kernel size (3, 4, and 5 respectively) and relu activation function with 128 filters and a single layer of the LSTM network. Followed by the pooling, dense layer with relu activation function, dropout layer and final dense layer with softmax activation function. Figure 4 shows this architecture where the input is directed to an embeding layer and then the layered CNN and LSTM. This model is compiled with the catagorical_crossentropy loss function, Adam optimizer, and with learning rate of 0.00001.

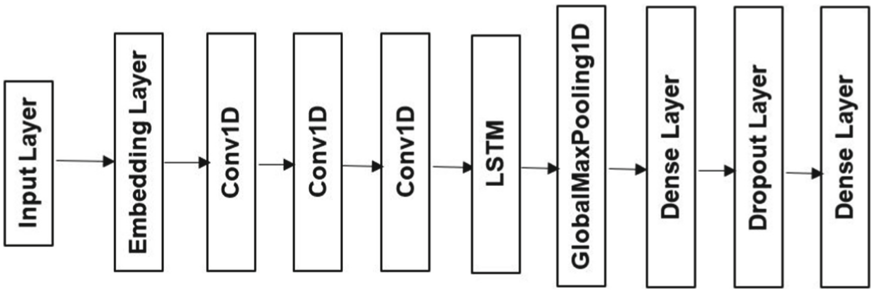

Fig. 4. CNN-LSTM architecture.

4.5 Distiled Bidirectional Encoder Representation from Transformer (DistilBERT)

Bidirectional Encoder Representation from Transformers (BERT) is a fine-tuning-based deep bidirectional language representation model. The BERT model is pre-trained on unlabeled data over different pre-training tasks [6]. In this study, we use DistilBERT, a general-purpose pre-trained version of BERT that retains 97% of the language understanding capabilities [27]. DistilBERT uses knowledge distillation during the pre-training phase to reduce the model's size. We fine-tune the model using the distilbert-base-uncased transformers model from Huggingface [33]. The model includes the input layer with input ids and attention mask, followed by the distilBERT main layer, global max pool layer, dropout and final dense layer. The model uses 0.00002 as learning rate with Adam optimizer and categorical_crossentropy loss function. (refer Fig. 5).

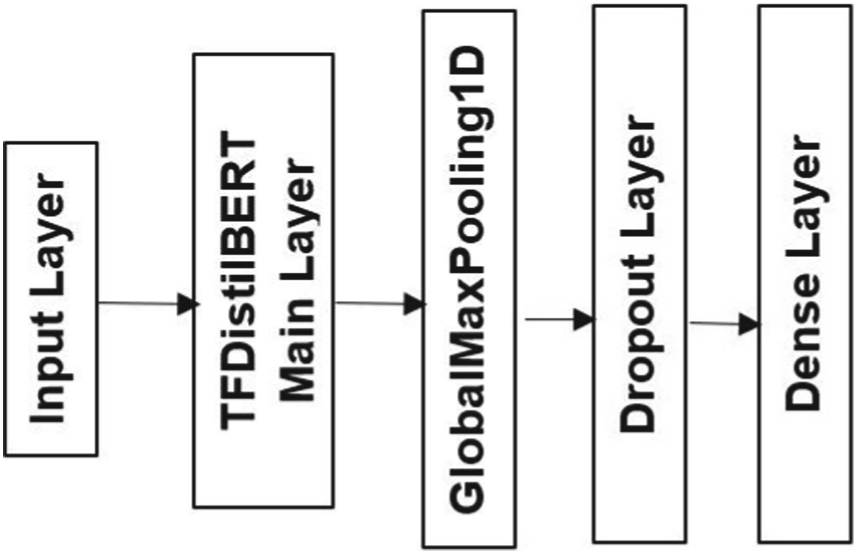

Fig. 5. DistilBERT architecture.

4.6 Stratified K-Fold Cross Validation

In regular k-fold cross-validation, the data is partitioned into approximately k equal folds. Train and validation is performed on the partitioned data for 'k' iterations. In each iteration, the k − 1 fold is used for training the model and the remaining one fold for validation. The accuracy obtained in each iteration is then averaged to get the overall model accuracy. In stratified k-fold cross validation, the data is arranged in such a way that each fold is a good representation of the whole dataset. In this paper, we use 5-fold stratified cross-validation. Further details are described in Sect. 5.

5 Experiments and Results

The proposed models are trained and tested on Google Colab. We use the dataset from Sect. 3 and apply the tokenization and padding from Sects. 4.1, for the models. To train and evaluate the models, we use stratified k-fold cross-validation with k = 5. The model performance is shown using the average validation accuracy of the final epochs from each of the five folds in the cross-validation. For example, the CNN-LSTM model was trained for ten epochs. The validation accuracy that is calculated on the 10th epoch is the one that was utilized to calculate the average validation accuracy.

As it can be seen in Table 3, the DistilBERT model achieved the highest accuracy of 91.46%. This may be due to the fact that attention masks were added to the input of the DistilBERT model along with the input ids. The attention masks allow the layers of the model to differentiate between the tokens that are the results of padding and the ones that are not. The next best-performing model was the CNN model with an accuracy of 85.17%, followed by the CNN-LSTM model with an accuracy of 84.30%. The LSTM model achieved an accuracy of 83.96%. It can be observed that CNN has better performance than LSTM as the two models with the least accuracies involve LSTM layers.

Table 3. Average accuracies and hyperparameters.

Model	Average accuracy	#Epochs
Convolutional Neural Network (CNN)	85.17%	20
Long Short Term Memory (LSTM)	83.96%	30
CNN-LSTM	84.30%	10
DistilBERT	91.46%	2

6 Conclusions

Public health emergencies provide a drastic change in peoples day to day activities. It is essential to monitor such situations to mitigate future emergencies. Social media platforms are major source of information disemination all over the world. People share their opinion and feeling in every aspect. Especially the pandemic and the non-pharmacheutical intervention heightened public emotions and distress. To capture people's emotions and to understand the spread of disease and public health protocols, it is important to process such information using computational models, especially because of the complex nature of the data. Natural language processing and sentiment analysis help us understand such public perceptions. We collected data using covid-related hashtags and presented a deep learning model framework for the sentiment classification of tweets into positive, negative, and neutral classes. We observe the best performance of 91.46% with the DistilBERT model. We use stratified 5-fold cross-validation, and the average accuracy obtained is shown in Table 3 for all the models. In this digital era, big data is obvious in every domain. Especially social media data, with millions of users all around the world. The transformer distilBERT model is computationally complex. In future we plan to integrate distributed computing using Spark for the transformer model to improve the computational time for big data sets.

References

1. Agichtein, E., Castillo, C., Donato, D., Gionis, A., Mishne, G.: Finding high-quality content in social media. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, pp. 183–194 (2008)

2. Alamoodi, A., et al.: Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: a systematic review. Expert Syst. Appl. **167**, 114155 (2021)
3. Albawi, S., Mohammed, T.A., Al-Zawi, S.: Understanding of a convolutional neural network. In: 2017 international Conference on Engineering and Technology (ICET), pp. 1–6. IEEE (2017)
4. Chakraborty, K., Bhatia, S., Bhattacharyya, S., Platos, J., Bag, R., Hassanien, A.E.: Sentiment analysis of COVID-19 tweets by deep learning classifiers-a study to show how popularity is affecting accuracy in social media. Appl. Soft Comput. **97**, 106754 (2020)
5. De Choudhury, M., Counts, S., Czerwinski, M.: Identifying relevant social media content: leveraging information diversity and user cognition. In: Proceedings of the 22nd ACM Conference on Hypertext and Hypermedia, pp. 161–170 (2011)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
7. Dos Santos, C., Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pp. 69–78 (2014)
8. Drus, Z., Khalid, H.: Sentiment analysis in social media and its application: systematic literature review. Proc. Comput. Sci. **161**, 707–714 (2019)
9. Feldman, R.: Techniques and applications for sentiment analysis. Commun. ACM **56**(4), 82–89 (2013)
10. Gao, Z., Feng, A., Song, X., Wu, X.: Target-dependent sentiment classification with BERT. IEEE Access **7**, 154290–154299 (2019)
11. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Proj. Rep. Stanford **1**(12), 2009 (2009)
12. Goularas, D., Kamis, S.: Evaluation of deep learning techniques in sentiment analysis from Twitter data. In: 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), pp. 12–17. IEEE (2019)
13. Gupta, P., Kumar, S., Suman, R., Kumar, V.: Sentiment analysis of lockdown in India during COVID-19: a case study on Twitter. IEEE Trans. Comput. Soc. Syst. **8**(4), 992–1002 (2020)
14. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
15. Imran, A.S., Daudpota, S.M., Kastrati, Z., Batra, R.: Cross-cultural polarity and emotion detection using sentiment analysis and deep learning on COVID-19 related tweets. IEEE Access **8**, 181074–181090 (2020)
16. Jianqiang, Z., Xiaolin, G., Xuejun, Z.: Deep convolution neural networks for Twitter sentiment analysis. IEEE Access **6**, 23253–23260 (2018)
17. Liao, S., Wang, J., Yu, R., Sato, K., Cheng, Z.: CNN for situations understanding based on sentiment analysis of twitter data. Proc. Comput. Sci. **111**, 376–381 (2017)
18. Loureiro, D., Barbieri, F., Neves, L., Anke, L.E., Camacho-Collados, J.: TimeLMs: diachronic language models from Twitter. arXiv preprint arXiv:2202.03829 (2022)
19. Mansoor, M., Gurumurthy, K., Prasad, V., et al.: Global sentiment analysis of COVID-19 tweets over time. arXiv preprint arXiv:2010.14234 (2020)
20. Murray, D.G., Simsa, J., Klimovic, A., Indyk, I.: tf.data: a machine learning data processing framework. arXiv preprint arXiv:2101.12127 (2021)
21. Murthy, D.: Twitter. Polity Press Cambridge (2018)

22. Pota, M., Ventura, M., Catelli, R., Esposito, M.: An effective BERT-based pipeline for twitter sentiment analysis: a case study in Italian. Sensors **21**(1), 133 (2020)
23. Ranganathan, J., Hedge, N., Irudayaraj, A.S., Tzacheva, A.A.: Automatic detection of emotions in Twitter data: a scalable decision tree classification method. In: Proceedings of the Workshop on Opinion Mining, Summarization and Diversification, pp. 1–10 (2018)
24. Ranganathan, J., Tsahai, T.: Analysis of topic modeling with unpooled and pooled tweets and exploration of trends during COVID. Int. J. Comput. Sci. Eng. Appl. (IJCSEA) **11** (2021)
25. Ritonga, M., Al Ihsan, M.A., Anjar, A., Rambe, F.H., et al.: Sentiment analysis of COVID-19 vaccine in Indonesia using naïve bayes algorithm. In: IOP Conference Series: Materials Science and Engineering, vol. 1088, p. 012045. IOP Publishing (2021)
26. Rouvier, M., Favre, B.: SENSEI-LIF at SemEval-2016 task 4: polarity embedding fusion for robust sentiment analysis. In: SemEval@ NAACL-HLT, pp. 202–208 (2016)
27. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. ArXiv abs/1910.01108 (2019)
28. Shofiya, C., Abidi, S.: Sentiment analysis on COVID-19-related social distancing in Canada using Twitter data. Int. J. Environ. Res. Public Health **18**(11), 5993 (2021)
29. Singh, R., Singh, R., Bhatia, A.: Sentiment analysis using machine learning technique to predict outbreaks and epidemics. Int. J. Adv. Sci. Res **3**(2), 19–24 (2018)
30. Sitaula, C., Shahi, T.B.: Multi-channel CNN to classify nepali COVID-19 related tweets using hybrid features. arXiv preprint arXiv:2203.10286 (2022)
31. Stojanovski, D., Strezoski, G., Madjarov, G., Dimitrovski, I.: Twitter sentiment analysis using deep convolutional neural network. In: Onieva, E., Santos, I., Osaba, E., Quintián, H., Corchado, E. (eds.) HAIS 2015. LNCS (LNAI), vol. 9121, pp. 726–737. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19644-2_60
32. Villavicencio, C., Macrohon, J.J., Inbaraj, X.A., Jeng, J.H., Hsieh, J.G.: Twitter sentiment analysis towards COVID-19 vaccines in the Philippines using naïve bayes. Information **12**(5), 204 (2021)
33. Wolf, T., et al.: Transformers: state-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.emnlp-demos.6, https://aclanthology.org/2020.emnlp-demos.6
34. Zhang, S., Zhang, X., Chan, J.: A word-character convolutional neural network for language-agnostic twitter sentiment analysis. In: Proceedings of the 22nd Australasian Document Computing Symposium, pp. 1–4 (2017)