Neural Processing Letters (2023) 55:2265–2304
https://doi.org/10.1007/s11063-022-11112-0

AI Assisted Attention Mechanism for Hybrid Neural Model to Assess Online Attitudes About COVID-19

Harnain Kour[1] · Manoj K. Gupta[1]

Accepted: 10 December 2022 / Published online: 23 December 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

COVID-19 is a novel virus that presents challenges due to a lack of consistent and in-depth research. The news of the COVID-19 spreads across the globe, resulting in a flood of posts on social media sites. Apart from health, social, and economic disturbances brought by the COVID-19 pandemic, another important consequence involves public mental health crises which is of greater concern. Data related to COVID-19 is a valuable asset for researchers in understanding people's feelings related to the pandemic. It is thus important to extract the early information evolving public sentiments on social platforms during the outbreak of COVID-19. The objective of this study is to look at people's perceptions of the COVID-19 pandemic who interact with each other and share tweets on the Twitter platform. COVIDSenti, a large-scale benchmark dataset comprising 90,000 COVID-19 tweets collected from February to March 2020, during the initial phases of the outbreak served as the foundation for our experiments. A pre-trained bidirectional encoder representations from transformers (BERT) model is fine-tuned and embeddings generated are combined with two long short-term memory networks to propose the residual encoder transformation network model. The proposed model is used for multiclass text classification on a large dataset labeled as positive, negative, and neutral. The experimental outcomes validate that: (1) the proposed model is the best performing model, with 98% accuracy and 96% F1-score; (2) It also outperforms conventional machine learning algorithms and different variants of BERT, and (3) the approach achieves better results as compared to state-of-the-art on different benchmark datasets.

Keywords COVID-19 · Transfer learning · Tweets · Sentiment analysis · LSTM · BERT

Abbreviations

BERT	Bidirectional encoder representations from transformers
LSTM	Long short-term memory network
RETN	Residual encoder transformation network
SARS-CoV-2	Severe acute respiratory syndrome - Coronavirus

✉ Harnain Kour
18dcs008@smvdu.ac.in

[1] Department of Computer Science and Engineering, Shri Mata Vaishno Devi University, Katra, India

WHO	World Health Organization
NLP	Natural language processing
RNN	Recurrent neural network
NB	Naïve Bayes
RCNN	Recurrent convolution neural network
RF	Random forest
CNN	Convolutional neural network
RMSProp	Mean square propagation
DL	Deep learning
ML	Machine learning
SVM	Support vector machine
AI	Artificial intelligence
DT	Decision tree
HyRank	Hybrid ranking
EPF	Emotional proximity function
DAN	Deep averaging network
ETC	Extra trees classifiers
IWV	Improved word vector
KNN	K-nearest neighbour
LR	Logistic regression
GloVe	Global vectors for word representation
Bi-LSTM	Bidirectional LSTM
GAME	Graph attention model embedded
Acc	Accuracy
Pre	Precision
Adagrad	Adaptive gradient
CRF	Conditional random field
TN	True negative
SGD	Stochastic gradient descent
CGAN	Conditional generative adversarial network
PV-DM	Distributed memory paragraph vector
SBERT	Sentence-BERT
P	Actual positive
USE	Universal sentence encoder
ANNs	Artificial neural networks
N	Actual negative
FN	False negative
Adam	Root adaptive moment estimation
FP	False positive
MSE	Mean squared error
Rec	Recall
TP	True positive
F1	F1-score
SA	Sentiment analysis

1 Introduction

The COVID-19 disease derived from Severe Acute Respiratory Syndrome – Coronavirus (SARS-CoV-2) has spread across the world and have a significant impact on both population and the global economy. It was declared a pandemic by the World Health Organization (WHO) on January 30, 2020 [1]. In 2020, the global COVID-19 pandemic sparkled an unprecedented public health crisis. There has been a tremendous influence on mental stress due to the problems that the virus brought, like loss of employment, depression, loneliness, lack of income, home-schooling, and many more. Therefore, the adverse effects of the coronavirus disease have drastically impacted social, professional, and personal lives all over the world. Additionally, there were restrictions imposed on people's movements, which has disrupted important day-to-day work. To control the spread of virus and to save the public from contracting it, the governments of most of the affected nations imposed a strict quarantine for different periods in order to break the chain of this pandemic. This provided an ideal opportunity for people to express and share their opinions on social networking sites. Therefore, amid the isolation, people used to spend more time on social networking sites to calm themselves, find information, and express their feelings [2]. In addition, they are more likely to trust information found online, such as when to get tested, where to seek care, and what to do with the test results. However, the social media information can often be incorrect or misleading, which can cause stress, anxiety, and depression [3]. Therefore, people's pain gets worsened when they come across such misleading and depressing news on social media. As a result, the imposed COVID-19 restrictions severely affect the population's mental as well as physical health [4]. Although the physical symptoms of this virus are widely known, this study focused on some of its unfamiliar influences, such as adverse mental health effects. The government occasionally conducts surveys to see how individuals view the pandemic in the context of sentiments related to their mental health, but this approach is laborious, expensive, and time-consuming. Even with the survey, a very small portion of the population can be reached, and the sample size is too small for prediction and to aid medical professionals in making accurate diagnoses, and assist them come up with solutions for ending this pandemic.

Social media has provided researchers access to large scale free data and has played a significant role in motivating them to investigate public attitudes towards COVID-19 during the pandemic. It has been observed that during the quarantine, traffic on social networking increased tremendously [5], resulting in 3.96 billion social media users worldwide in 2021 as shown in Fig. 1. Twitter is the most popular social media platform, where individuals reflect real-time public sentiments via 140-character tweet. The number of daily tweets has also increased by 23% during the COVID-19 pandemic in 2020 [6, 7]. Thus, the increased use of Twitter by people during the lockdown period motivated us to investigate Twitter data to assess human sentiments regarding the COVID-19 pandemic. Therefore, it becomes the most important platform for researchers to investigate insights from an individual's psychology, i.e., the current mental state, attitude, behavior, and so on. However, social media posts contain a lot of unstructured and noisy data, therefore extracting relevant information from online data is a challenging task. As a result, it necessitates the study of different Deep Learning (DL), Machine Learning (ML), and Natural Language Processing (NLP) technologies commonly used in Artificial Intelligence (AI) applications [8].

The use of NLP and its applications in social media analysis has grown at an alarming rate. Most of the recent NLP technologies are prone to adversarial texts [9]. Also, mining the underlying meaning of a text using NLP techniques is challenging owing to its unstructured

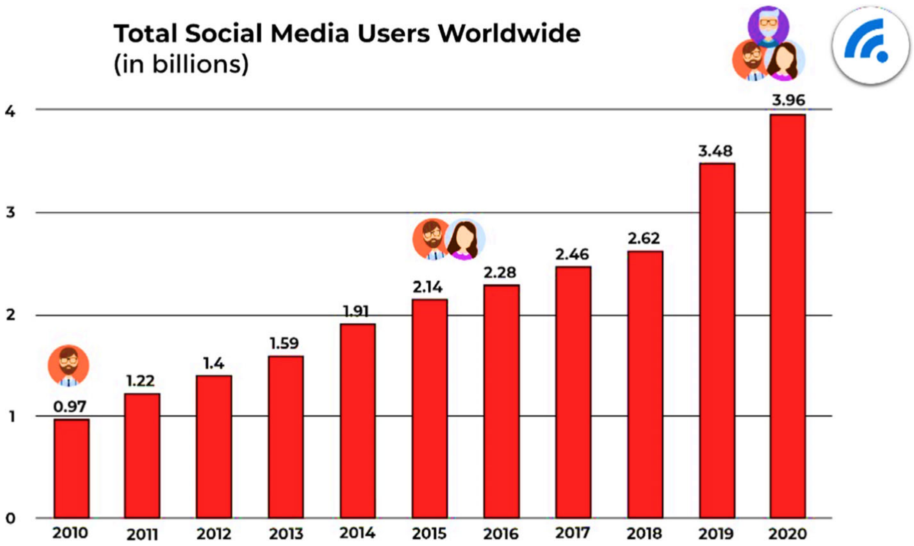


Fig. 1 Social media users in the world: 2021 [7]

form. Moreover, it has been observed that neural network models perform poorly in NLP as compared to computer vision tasks. It is because most of the text datasets are too small to train DL models, which contain a large number of parameters and overfit when trained on such small datasets [10]. Another significant reason for NLP's lag beside computer vision is the absence of transfer learning in NLP. Further, DL in computer vision has greatly benefited through transfer learning owing to the accessibility of large-scale labeled datasets, viz ImageNet, on which DL models were trained and then utilized as pre-trained models for various computer vision tasks [11]. Since Google introduced the transfer learning model in 2018, NLP has aided in the accomplishment of a variety of tasks with cutting-edge performance. Hence, it is crucial to understand the limitations of various text categorization approaches as well as related ML algorithms in comparison to pre-trained transfer learning models [9]. The DL advancements have resulted in neural network models such as Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and Long Short-Term Memory (LSTM) network showing promising improvements in tackling several NLP tasks such as language modeling, machine translation, text classification, and so on. RNNs come in a variety of forms, such as basic RNNs, LSTMs, and Gated Recurrent Units (GRUs). Each DL-based architecture has a unique set of pros and cons. GRU employs gates to decide whether or not to forward the prior input to the subsequent cell, whereas simple RNN lacks gates. In addition, the GRU network contains a memory unit. In contrast, LSTM is more effective since it contains two more gates, the "Forgot gate" and the "Output gate". Moreover, the LSTM incorporates a feedback loop that aids in the processing of both sequential and single data points [12]. Even biLSTM models are ineffective in capturing the real meaning of words because the true context is lost as they learn the “right-to-left” and “left-to-right” contexts separately before combining them. Further, biLSTM models can perform noticeably better than Bidirectional Encoder Representations from Transformers (BERT) models for a smaller dataset and can be trained in less time than transfer-learning-based pre-trained models [13]. However, these transfer-learning-based pre-trained models can be fine-tuned to solve deep learning's shortcomings by perceiving actual context while simultaneously learning how to tackle a specific task [14]. All these factors have been taken into consideration while choosing a model because the performance of the model depends on both data and task.

In this paper, various state-of-the-art ML techniques are used to obtain appropriate results and compared them with transfer learning techniques for COVID-19 Sentiment Analysis (SA) using the Twitter dataset. We also developed a hybrid model namely, Residual Encoder Transformation Network (RETN) by combining neural networks i.e., two LSTM networks, and a transformer-based language model i.e., BERT to overcome these limitations. Among the most significant contributions of this study is the evaluation and comparison of text sentiment classification methods. The process has the ability to promote the use of AI in analyzing social interaction by extracting insights from the text that can recognize public sentiment and predict the outbreak of COVID-19 related fear.

The objective of this study is to answer three research questions:

RQ1 How to analyze people's sentiments expressed on Twitter as a result of COVID-19?

RQ2 How do different ML and transfer learning models perform in comparison to the proposed hybrid model on both individual and the combined COVIDSenti datasets?

RQ3 Is it possible to enhance the performance of sentiment classification techniques by feature engineering and using hybrid models?

To address these questions four benchmark datasets were created by Usman Naseem et al. [15], namely, COVIDSenti-A, COVIDSenti-B, COVIDSenti-C, and COVIDSenti are utilized for performing SA on COVID-19 pandemic related tweets. All of the four COVIDSenti datasets have been categorized into three sentiment classes, negative, neutral, and positive. Moreover, a hybrid architecture, namely, RETN is proposed in this study that uses two strategies. Firstly, it is trained using a pre-trained transfer learning model i.e., BERT on all the four COVIDSenti datasets. Secondly, the embeddings extracted from BERT are combined with neural network architectures i.e., two LSTM networks. To draw a conclusive comparison of performance for the proposed RETN model for COVID-19 sentiment classification four commonly used state-of-the-art ML algorithms are used, namely, Support Vector Machine (SVM) [16], Decision Tree (DT) [17], Random Forest (RF) [18], and Naïve Bayes (NB) [19]. Moreover, the final results are also compared with transformer-based language architectures i.e., BERT [20, 21], DistilBERT [22], and RoBERTa [23].

1.1 Major Contributions

The following are the major contributions of this research:

(1) Proposed a hybrid model i.e., RETN based on multi-headed attention that uses residual connections between layers. The proposed model is a hybrid of transformer-based language model i.e., BERT and neural network architecture i.e., two LSTMs.
(2) Performance of various state-of-the-art ML text classifiers and BERT model variants is compared to RETN, and baseline results for each are evaluated using various performance metrics. Moreover, the results are also validated using three benchmark datasets, and the Friedman rank test supported the conclusions.
(3) Qualitative analysis and in-depth examination of public sentiments concerning COVID-19 are conducted using various visualizations of the content that is present in textual data.

(4) The key features of the COVIDSenti dataset are examined to identify and describe the dominant public discourse about COVID-19. Further, the study findings could assist government all over the world in developing effective public health policies.

The remaining part of this paper is framed as follows. The related work is presented in Sects. 2 and 3 discusses the dataset used. The proposed methodology that this paper will use is presented in Sect. 4. In Sect. 5, the results of the experiments employed in this study are compared and discussed in detail. The conclusion and highlights of the future work are presented in Sect. 6.

2 Related Work

The coronavirus has impacted the social and personal lives of individuals all around the globe. As a result, many researchers attempted to analyze sentiments about the COVID-19 pandemic from various perspectives and present their results in different ways using ML, DL, and NLP approaches. Li et al. [24] classified social media content using ML algorithms such as SVM, RF, and NB. Weibo, a microblogging site, was utilized to extract the 367,462 posts related to the COVID-19 pandemic, which were then classified into seven categories based on behavioral, perceptual, emotional, and affiliation factors. The average accuracies of SVM, NB, and RF classifiers were 54%, 45%, and 65%, respectively. The RF classifier outperformed the other classifiers and produced the best results. Baker et al. [25] analyzed Arabic tweets about the pandemic to determine the sentiments of the online users. The Twitter API was used to collect 54,065 tweets, which were then categorized using four classifiers, i.e., KNN, DT, NB, and SVM. The experiments revealed outstanding results, with KNN and NB achieving accuracies of 86.43% and 89.06%, respectively. Jianqiang and Xiaolin [26] proposed a hybrid Representation Global Vectors for Word Representation (GloVe)-Deep Convolutional Neural Network (DCNN) technique. The GloVe method evaluates the hidden context and semantic relationships between various words present in a phrase. Furthermore, the generated word embeddings were then combined with n-gram and word mood polarity score attributes to create a feature set. Finally, the outcome is fed to a DCNN model. The proposed GloVe-DCNN model performed best, with an accuracy of 87.62%, precision of 87.60%, recall of 87.45%, and an F1-score of 87.50%. Rustam et al. [27] developed a hybrid feature extraction method that incorporates Bag-of-Words (BoW) and Term Frequency (TF)-Inverse Document Frequency (IDF) methods to extract significant features from the text. The Twitter data was used to assess COVID-19 sentiments expressed by users in tweets. The extracted features were fed to five ML-based models, i.e., XGBoost, RF, SVC, DT, and ETC. Furthermore, 80% of the data comprising 6022 tweets was used for training, with the remaining 20% of the data comprising 1506 tweets was used for testing. The proposed hybrid feature set with Extra Trees Classifiers (ETC) surpassed all other models attaining an accuracy of 93%.

In another study, Naseem et al. [15] developed a large-scale benchmark dataset, namely, COVIDSenti, composed of 90,000 COVID-19 tweets gathered between February and March 2020, during the pandemic's initial phases. The tweets were categorized into three classes, i.e., negative, positive, and neutral. Conventional methods such as Word2Vec and TF-IDF, as well as word embedding-based methods such as fastText, GloVe, and Word2Vec, were utilized for extracting important contextual features. Moreover, for classification purposes, two hybrid models were proposed, namely, Improved Word Vector (IWV) and Hybrid Ranking (HyRank). In addition, transfer learning architectures such as DistilBERT, ALBERT, XLNET, and BERT were employed for classification. It was concluded that BERT outperformed all other hybrid and transfer learning models with an accuracy of 94.8%. Chintalapudi et al. [28] gathered tweets between March 23, 2020 and July 15, 2020 in order to examine people's emotions. Tweets were classified into four emotional categories, i.e., fear, sadness, joy, and rage. The transformer-based model BERT was used for experimentation, and its performance was compared to ML and DL approaches such as SVM, Logistic Regression (LR), and LSTM. The BERT model outperformed the other three models, i.e., SVM, LR, and LSTM, with an accuracy of 89%. To recognize the behavior and psychology of the public, Raamkumar et al. [29] proposed a Health Belief Model (HBM) to identify social media content prior to and after a pandemic. Moreover, the neural network-based text classification approaches were examined using online content prior-to and ahead-of the COVID-19 pandemic. The HBM text categorization models achieved a maximum accuracy of 95%. In another study, to perform text classification using the COVID-19 dataset, Sunagar et al. [30] employed DL models such as RNN, CNN, Recurrent Convolution Neural Network (RCNN), RNN-LSTM, and RNN-biLSTM. All models also employ two-word embedding approaches, namely, GloVe, and Word2Vec for feature extraction. The proposed hybrid RNN-biLSTM model outperformed all other classifiers with an accuracy of 93%.

Recently, Anjir et al. [12] utilized transfer and DL models, namely, LSTM, RNN, BERT, and GRUs to examine people's sentiments against COVID-19 vaccines. They carried out a questionnaire amongst the general public, and the data was collected via a Google form. When the results of all the models were compared, BERT produced the best results with the accuracy of 86%, precision of 83%, recall of 85% and F1-score of 84%. However, among DL models, LSTM outperformed RNN and GRU with the accuracy of 79%, precision of 83%, recall of 80% and F1-score of 81%. To perform the binary text classification, Qasim et al. [31] used nine transfer learning models, i.e., BERT-large, BERT-base, XLM-RoBERTa-base, RoBERTa-large, RoBERTa-base, DistilBERT, Electra-small, ALBERT-base-v2, and BART-large. The experiments used three datasets, namely, the COVID-19 Twitter dataset, the COVID-19 false news dataset, and the extremist-non-extremist dataset. BERT-large and BERT-base surpassed the existing approaches with an accuracy of 99.71%. Jaiman et al. [32] developed the COVID-TWITTER-BERT (CT-BERT), which was a BERT model pre-trained on 160 million COVID-19 tweets. CT-BERT performed well on a variety of Twitter datasets used for SA, as well as datasets associated with COVID-19 tasks. CT-BERT performed well, with 96.1% accuracy on the Twitter dataset related to COVID-19 and 99.3% accuracy on the Twitter dataset for SA. Basiri et al. [33] developed a fusion model based on a hybrid DL technique for SA utilizing COVID-19 tweets. In this study, the SA was performed on the Twitter data collected from eight different nations, like United States, Canada, Australia, Iran, England, China, Italy, and Spain. The proposed hybrid model was the fusion of the fast text model, CNN, NB-SVM, BERT, and Bi-directional Gated Recurrent Network (biGRU). Moreover, the proposed DL-based fusion model beat baseline techniques, namely, CNN, biGRU, fastText, NBSVM, and DistilBERT, with 85.5% accuracy on the Stanford sentiment-140 Twitter dataset.

In this paper, the SA is performed using COVID-19 tweets to find the most common sentiments of the public, which is similar to all preceding works. However, the majority of previous works have used dataset with one characteristic only. In contrast, we have utilized four different datasets containing COVID-19 tweets of varying characteristics to evaluate the model's ability to achieve the best classification performance. To reduce noise from tweets and create a balanced dataset, data pre-processing, augmentation, and oversampling techniques are used. Furthermore, BERT is used to produce a significant feature set with the aim of enhancing the performance. The extracted feature set is used with multi-LSTMs for the extraction of sequential features and sentiment classification. The performance of COVID-19 Twitter SA is compared using ML, transfer learning and the proposed hybrid model. Further, the benchmarked findings are evaluated to confirm the performance of the proposed hybrid RETN model using various benchmark datasets. In contrast to the existing methods, the proposed system effectively assesses the sentiments of COVID-19 tweets and extracts the most relevant feature sets, which aids in improving classification results.

2.1 Analysis Table

In this section, the recent studies that used machine, deep, and transfer learning techniques to perform SA using text data have been examined and discussed. The related recent studies (2020–2022) are summarised in Table 1.

3 Dataset Description

We used four large-scale benchmark Twitter datasets developed by Usman Naseem et al. [15], namely COVIDsenti, COVIDSenti-A, COVIDSenti-B, and COVIDSenti-C. Three datasets i.e., COVIDSenti-A, COVIDSenti-B, and COVIDSenti-C are combined to create the COVIDSenti dataset. Each of the individual datasets, COVIDSenti-A, COVIDSenti-B, and COVIDSenti-C, contains approximately 29,957 tweets, while the COVIDsenti dataset contains 90,450 tweets. The statistical information of the datasets used in this study are shown in Table 2. Each of the four datasets is comprised of three sentiments that have been categorized as positive, negative, or neutral for classification purposes. Some of the labeled tweets from the COVIDsenti dataset are shown in Table 3. Each of the four datasets has distinct characteristics, thereby providing a more precise comparison of the performance of various models utilized in this study. The following is a brief discussion of the characteristics of four datasets:

- COVIDSenti: This dataset contains 90,000 tweets gathered from 70,000 Twitter users from February 2020 to March 2020. The COVIDSenti dataset is the combination of three datasets as under:
- COVIDSenti-A: This dataset contains 30,000 unique tweets, out of which 1968, 5083, and 22,949 have been labeled as positive, neutral, or negative. This dataset contains the tweets about the government's actions and plans to tackle the COVID-19 pandemic. For e.g., "It's crazy to think that the government might have created the coronavirus and now scientists have created a vaccine https://t.co/sQWBCOpPK2".[1].
- COVIDSenti-B: This dataset contains 30,000 unique tweets, out of which 2033, 5471, and 22,496 have been labeled as positive, negative, or neutral. This dataset contains tweets primarily about the COVID-19 crisis, imposed quarantine, stay at home, and social distancing. As a result, it primarily addresses the periodic shift in people's behavior depending on the number of incidents, panic, terror-provoking information, etc. For e.g.., "thinking about how things like ebola & coronavirus have always made my anxiety so bad and now it's possibly less than https://t.co/ttIhRlkInQ". (See Footnote 1).
- COVIDSenti-C: This dataset contains 29,997 unique tweets, out of which 2276, 5781, and 21,940 have been labeled as positive, negative, or neutral. This dataset contains tweets about COVID-19 cases, breakouts, and lockdowns, thereby revealing some aspects of

[1] This is a real Twitter post that is included in our dataset

Table 1 Analysis of recent studies (2020–2021)

References	Author and year	Dataset used	Classification techniques	Results	Weakness	Strength
[34]	Samuel et al. (2020)	COVID-19 Twitter dataset	NB, LR, and KNN	Accuracy (Acc) = 0.91	The article used single keyword monitoring to analyze the attitudes of U.S.A citizens Other element subjected to geographical restrictions need to be investigated Performance analysis based on traditional ML algorithms	Fast recompilation on new data points
[35]	Priya et al. (2020)	Questionaries	SVM, DT, RF KNN, and NB	Acc = 85 Recall (Rec) = 85 F1 = 83.6 Pre = 82	It is tough to confirm the efficiency and generality of their study on a huge dataset	The technique used is simple and efficient for classification concerning multiclass problems
[15]	Naseem et al. (2021)	COVID-19 Twitter dataset	ALBERT, DistilBERT, BERT, IWV, XLNET, and HyRank	Acc = 94.8	The proposed model must additionally be tested on other related datasets in order to verify the model's performance	The study has provided the in-depth information about the features utilized
[33]	Basiri et al. (2021)	COVID-19 Twitter dataset	CNN, NBSVM, DistilBERT, biGRU, and the novel fusion model	Acc = 85.5 F1-score (F1) = 85.5	It is important to evaluate performance of the proposed model by considering datasets with various characteristics during the COVID-19 pandemic	Investigated the performance of various models by validating on a huge twitter dataset consisting of 1600,000 tweets Analyzed data related to COVID-19 from various demographic locations

Table 1 (continued)

References	Author and year	Dataset used	Classification techniques	Results	Weakness	Strength
[36]	Kaur et al. (2021)	COVID-19 Twitter dataset	SVM, RNN, and Heterogeneous Support Vector Machine (H-SVM)	F1 = 95.4 Precision (Pre) = 69 F1 = 77 Acc = 96.3	There is a need to balance the imbalanced data before classification	The proposed model improved the overall classification performance
[37]	Rosario et al. (2021)	COVID-19 medical records	biLSTM-CRF (Conditional Random Field)	F1 = 92.11	It examined various versions of BERT to see which are most suitable for clinical de-identification scenarios	The biLSTM-CRF architecture was evaluated in combination with a layered embedding comprised of fastText and Flair embeddings
[38]	Chandrasekaran and Hemanth (2022)	COVID-19 Twitter dataset	NB, LR, SVM, biLSTM	Acc = 87	The proposed model is evaluated on one dataset only which makes the worth of the other models under doubt	Use of biLSTM neural architecture learns right-to-left and left-to-right contexts present in the sentence
[39]	Singh et al. (2022)	COVID-19 Twitter dataset	RNN-LSTM with attention mechanism	Acc = 86.12 Pre = 84.23 Rec = 85.23 F1 = 85.12	It is important to use feature extraction methods which may further improve the classification results	Use of Leaky ReLU influences forward and backward training in RNN-LSTM which effectively controls error
[40]	Sunitha et al. (2022)	COVID-19 Twitter dataset	Ensemble of GRU and CapsNet	Acc = 97.28	There is a need to enhance classification accuracy while keeping computational complexity to a minimum	The proposed architecture extracts discriminative feature vectors using feature extraction techniques with an ensemble of DL models

Table 2 Statistics of datasets used

Dataset	Positive	Neutral	Negative	Total
COVIDSenti	6280	67,835	16,335	90,450
COVIDSenti-A	1968	22,949	5083	30,000
COVIDSenti-B	2033	22,496	5471	30,000
COVIDSenti-C	2276	21,940	5781	29,997

Table 3 Sample tweets from COVIDSenti dataset

Tweets	Labels
CDC health advisory about the Chinese coronavirus outbreak: ask patients with severe respiratory disease about travâ€šÃ„Â¶ https://t.co/CHs83xt8Qy	Neutral
I am increasingly convinced that #coronavirus is a design virus… A design that will not be pleasant forâ€šÃ„Â¶ https://t.co/2TI9XeQL8A	Positive
Let’s focus on the coronavirus. Public health concerns apart the economic costs of China under a lockdown will be fâ€šÃ„Â¶ https://t.co/jTH7CN22ha	Neutral
For up to date legitimate information on the #Coronavirus please keep in touch with your local #healthcare providerâ€šÃ„Â¶ https://t.co/ldQFSUCvAN	Neutral
COVID-19 STILL NOT HERE IM STILL GETTIN FUCKED UP	Negative
Nebraska woman with coronavirus, aged 36, is in a critical condition https://t.co/KRmIYS3AUQ	Negative
@abdbozkurt The guy is talking about some non-proven concepts like, COVID-19 tends to effect Asians more and sharesâ€šÃ„Â¶ https://t.co/R6Pl34Tr1s	Positive
The coronavirus hysteria is going to be more annoying to me than the plastic bag issue of 2020 in @dwtncamdensc	Negative
Iâ€šÃ„Ã´m still more worried about getting herpes than getting the coronavirus	Positive
Announced dead: doctor who issued coronavirus alert still struggles to survive	Negative

human behavior concerning a rise in the number of COVID-19 cases. “UPDATE: The whole of Italy has been put on lockdown to stop the spread of Coronavirus which has claimed 366 lives. https://t.co/HZToVaDWdg”. (See Footnote 1)

4 Proposed Methodology

The proposed methodology is comprised of the following modules, i.e., Dataset pre-processing; Feature extraction; Comparative methods; Classification; Results and Discussion. The dataset pre-processing module utilizes GAN, NLP, and data augmentation techniques to clean unstructured and noisy data. Moreover, to prepare data for the following module, this module facilitates data balancing, data cleaning, and tokenizing of tweets. Further, the feature extraction module extracts relevant and important contextual features of text using word embedding methods such as Doc2Vector (Doc2Vec), Sentence-BERT (SBERT), and Universal Sentence Encoder (USE). Hereafter, the ML-based methods, i.e., RF, SVM, and NB, and transfer learning models i.e., BERT, RoBERTa, and DistilBERT are utilized to classify

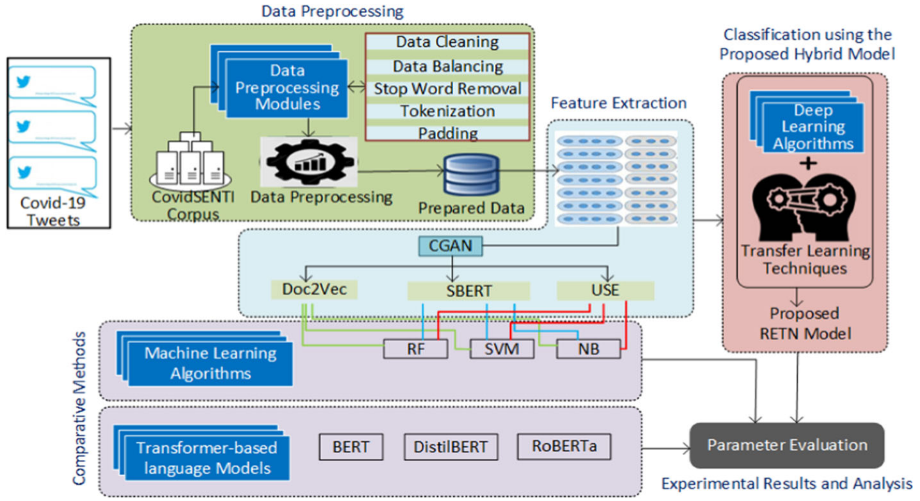


Fig. 2 Methodology layout

the sentiments present in the tweets. In addition, the transfer and deep learning-based hybrid model i.e., the proposed RETN model is utilized to categorizes the COVID-19 pandemic-related sentiments. Finally, in the last module, the results obtained from all the models are comparatively discussed concerning their accuracy, F1-score, precision, and recall. The layout of the proposed methodology is shown in Fig. 2.

In this paper, the sentiment classification is performed on four COVID-19 pandemic-related datasets and the comparative analysis is done utilizing the results obtained from traditional ML, state-of-the-art, and transfer learning models. Also, the performance is improved by extracting important features from tweets using the proposed RETN model and validated using three benchmark datasets. Moreover, the proposed hybrid model surpasses existing state-of-the-art models on all four COVID-19 pandemic-related Twitter datasets with higher accuracy and competence.

4.1 Data Pre-processing

4.1.1 Data Augmentation

Data Augmentation is commonly applied for computer vision tasks and it's challenging to use it for NLP tasks as augmentation techniques in NLP have not been fully examined [41]. The data augmentation techniques simply attempt to recreate the data distribution while preserving the semantics of all univariate distributions. This provides a very robust and simple mechanism for both data oversampling as well as undersampling. In this paper, the following data augmentation operations are performed [42]:

(a) **Synonym Replacement:** In this step, n words from the sentence are selected at random that are not stop words. Each of these words is substituted with a synonym chosen at random.

(b) **Random Insertion:** Random synonym for a random word is replaced in the statement that is not a stop word. Synonym is inserted at a random position in the statement and this step was repeated n times.

(c) **Random Swap:** Two words are selected in the statement at random and their positions are swapped. This step is repeated n times.

(d) **Random Deletion:** Each word in the statement is deleted at random with probability p.

4.1.2 Generative Adversarial Network (GAN)

A balanced dataset is created for resolving class imbalance problem using an oversampling technique. Oversampling is a common data creation methodology in which the dataset is not only fully utilized but is used to create a variant data source of the minority class. The probability distribution of the variant or new data that is augmented or generated has the same probability distribution as the parent class [43]. GANs were first demonstrated as a probability distribution function mapping neural network function to various autoencoders. The discriminator and generator functions collaborate to produce results [44]. One of the variants of conceptual probability distribution function generation is the conditional generative adversarial network (CGAN) [45]. This technology is primarily used to generate the probability distribution as well as class-wise kernel segregation.

In this paper, the CGAN technique is utilized for the text classification task that includes the conditional generation of text by a generator model. The typical CGAN architecture is shown in Fig. 3. In CGAN, the conditional setting is one in which both the generator and the discriminator are conditioned on certain types of information, such as class labels or data. The CGANs have an additional input layer to handle this additional information. This additional layer informs the generator about the class to predict. A feature matrix with a selection of distinctive textual characteristics serves as the input to this extra layer. As a consequence, while being given various contextual cues, the network concurrently learns the mapping from inputs to outputs. The sentence's probability vector is calculated for different noises, and the generator outputs a feature matrix. The output feature matrix from the generator and the feature matrix of the real data sample is used to train the discriminator. Finally, the discriminator shows the probability of real or fake data. We created a probability distribution of 0.2% for all the COVIDSenti datasets. Furthermore, 0.3, 0.3, and 0.1% distributions are

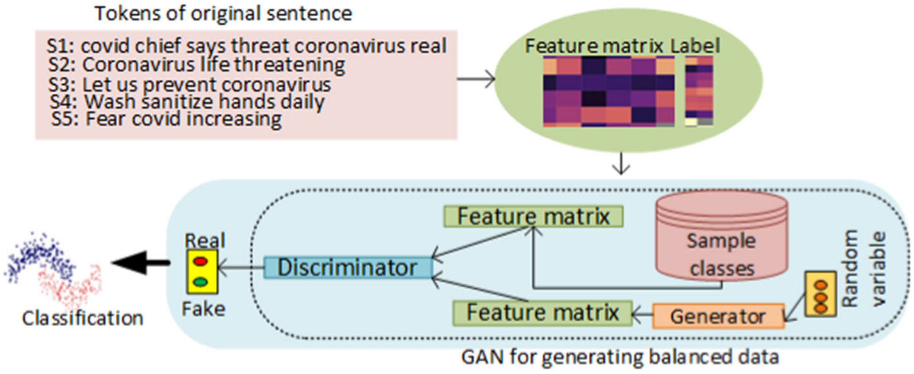

Tokens of original sentence
S1: covid chief says threat coronavirus real
S2: Coronavirus life threatening
S3: Let us prevent coronavirus
S4: Wash sanitize hands daily
S5: Fear covid increasing
Feature matrix Label
Feature matrix
Sample classes
Real
Fake
Discriminator
Feature matrix
Generator
Random variable
Classification
GAN for generating balanced data

Fig. 3 Data balancing using GAN

created for tweets labeled as positive, negative, and neutral. This provides a framework for selecting and generating features for classification models. This percentile change in the feature convolution of the data by GAN aids the model in discriminating the classes more effectively in higher feature spaces. As a result, the model's ability to comprehend the problem's situation is far more likely than its previous ancestral models.

4.1.3 Data Cleaning

Once the data set has been processed using the aforementioned techniques, various NLP techniques are used as discussed below:

- **Removal of irrelevant data:** In this step, the unstructured text patterns defined by the user in each tweet are eliminated. The unstructured text patterns comprise of, "user handles (@username)"; "hashtags (#hashtag)"; "characters, symbols, and numbers other than alphabets"; "empty strings"; "rows with NaN in the column"; "duplicate rows" and so on. Further, URLs are not considered in this study since they are not relevant for prediction and omitting them reduces computational complexity.
- **Tokenization:** Tokenization of all filtered tweets in a dataset is an important step in any text classification problem. It is the method that divides a long text string into small tokens and assigns a numerical representation to each token [46]. At last, a volumetric measure of the corpus is generated, which can be fed into the ML or DL-based network.
- **Stop word removal:** Stop words [47] are a group of terms that are regularly used in a language. Stop words include words such as, "the", "is", "a", "are", and so on. Removal of such words can result in the exclusion of low-level information present in the text. However, to avoid the loss of crucial information, we did not remove stop words.

4.2 Feature Extraction

Text pre-processing is followed by feature extraction to generate a feature set or word embedding consisting of high-quality features that are then fed to different ML algorithms. Therefore, three distinct word embedding techniques, namely, Doc2Vec, SBERT, and USE are used to extract significant and vital features from COVID-19 tweets.

4.2.1 Doc2Vector (Doc2Vec)

Doc2Vec [48] technique uses word tokens to extract semantic similarities between words that sound similar. In this study, the distributed memory paragraph vector (PV-DM) model is used as Doc2Vec model as shown in Fig. 4. For PV-DM, the Genism Doc2Vec class parameter "dm" is set to 1. DM is like a continuous bag of words from which interns create a dynamic sliding window and attempt to capture the relevant meaning of the words by sliding it through the sentence and the entire word corpus. It functions as a memory network that extracts the current context and changes the topic from one paragraph to the next [49]. Following tokenization, a single paragraph is created which comprises of the tokenized words as well as the appropriate tag. For PV-DM, the dimensionality of the feature vector is fixed at 300 × 1, whereas negative word sampling is permitted up to 5 with a minimum word count of 2. Hierarchical SoftMax is the activation function utilized in this sort of arrangement.

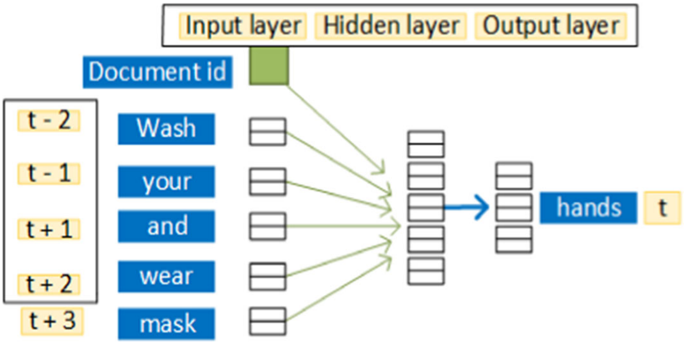


Fig. 4 Distributed memory model

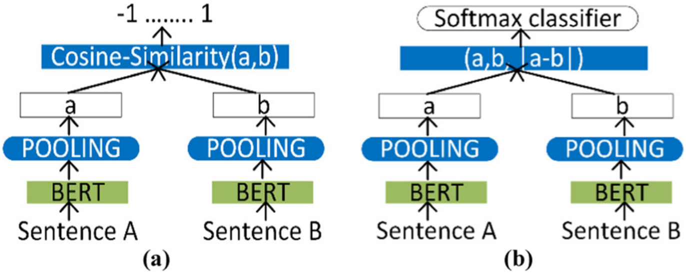


Fig. 5 Sentence-BERT **a** Regression Task **b** Classification task

4.2.2 Sentence-BERT (SBERT)

SBERT approach encodes sentences into a dense feature space of fixed-size, allowing semantically identical sentences to be positioned close to each other. SBERT [50] is based on transformer models like BERT [21] which encode each sentence independently and map them to a dense vector space. In this study, we used SBERT as a sentence-Transformer-to-vector architecture. Further, SBERT generates a sentence embedding of fixed-sized by incorporating a pooling operation to the BERT output. The default configuration is set to MEAN. BERT is fine-tuned using siamese and triplet networks [51] to adjust the weights so that the sentence embeddings generated are semantically relevant. In the classification task, sentence embeddings a and b is concatenated with the difference $|a - b|$ and then multiplied with the weight $W_t \in R^{3n \times k}$: $o = softmax(W_t(a, b, |a - b|))$. The dimension of the sentence embeddings is given by n, and the number of labels is given by k [50]. It is optimized using a cross-entropy loss function. The cosine similarity between the two sentence embeddings, a and b, is calculated in the regression task, as shown in Fig. 5a and b. Moreover, it is optimized using Mean Squared Error (MSE) loss which is utilized as the objective function.

4.2.3 Universal Sentence Encoder (USE)

The Universal Sentence Encoder [52] is a multi-task learning sentence encoding technique that is widely used for SA frameworks. In this study, the transformer Deep Averaging Network (DAN) is used. The DAN model can take in multivariate sentences and generate the

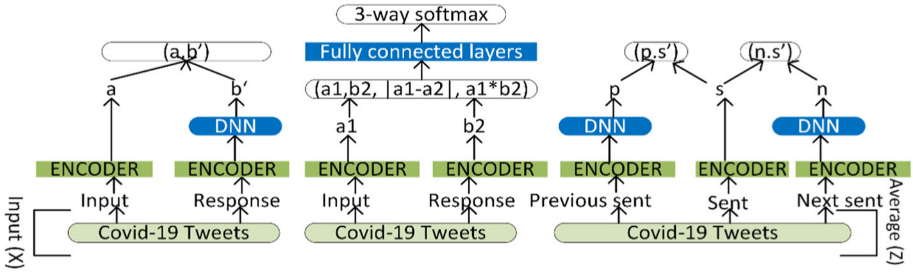


Fig. 6 Universal sentence encoder

corresponding embeddings. The sentence is first organized into lowercase dynamic token output sentences. Next, it is converted to a 512-dimensional vector and fed into a transformer with a self-attention mechanism. The DAN then computes and averages the unit unigram and bigram embeddings to obtain a univariate embedding. Following that, a deep neural network with a 512-feature space embedding of a final sentence length embedding processes the univariate embedding. The final data outcome can be used in a variety of supervised and unsupervised learning techniques. The average z is computed and the vector representation for input text X are created by averaging the word vectors $v_{w \in X}$. Instead of directly passing this representation to an output layer, z is transformed by adding more layers before applying the softmax [53]. Suppose we have n layers, $z_{1 \ldots n}$. Each layer is computed and the final layer's representation, z_n, which is fed to a softmax layer for prediction as shown in Fig. 6.

4.3 Proposed Hybrid (RETN- Residual Encoder Transformation Network) Model

The proposed hybrid RETN model employs transfer learning-based pre-trained BERT model to extract important word embeddings using COVID-19-related Tweets. The extracted word embeddings are integrated with multiple LSTM networks, thereby, achieving exceptional performance as compared to conventional ML methods, various BERT variants and state-of-the-art approaches. Thus, the BERT model functions both as a word embedding creator and the transfer learning backbone architecture for the proposed hybrid RETN model. The proposed model is divided into three modules: BERT, embedding batches, and the LSTM network.

4.3.1 BERT

BERT is a "deep bidirectional" model [22] which has been pre-trained on a huge corpus of unlabeled text, i.e., the entire Wikipedia (2500 million words) and BooksCorpus (800 million words). BERT has several variants, including RoBERTa, ELECTRA, DistilBERT, and ALBERT [22]. In this paper, DistilBERT [20] and RoBERTa [23] have been used for experimentation purposes, and their results are compared to the conventional ML classifiers, state-of-the-art techniques, and the proposed RETN model. The BERTBASE is used in this study via its retained uncased network [21, 54]. The detailed BERT architecture is presented in Fig. 7.

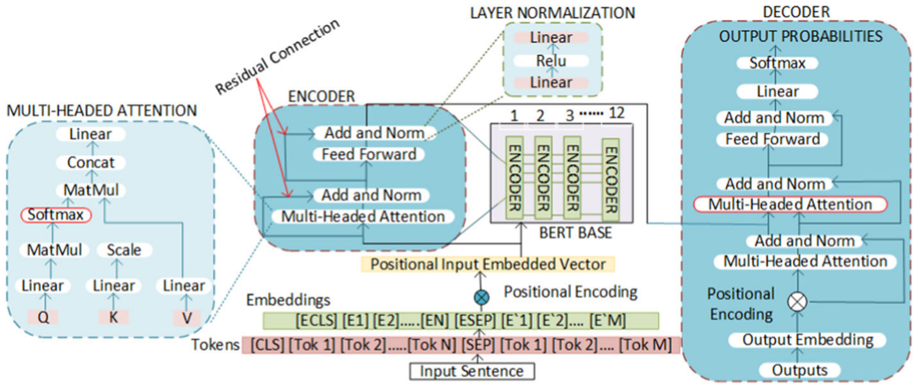


Fig. 7 BERT architecture

After tokenization, the input is fed into a word embedding layer such that x $\in R^{N,S}$. Here, x is the input, N is the batch size and S represents the length of encoder tokens. The encodings start with a $[CLS]$ token and tokens represent sub-words of the input text. Due to the lack of recurrence in the transformer's encoder, the positional information is incorporated within input embeddings. Next, is the encoder layer which consists of multi-headed attention, two sub-modules, and a fully connected network [21]. Moreover, there are additional residual connections encompassing both sublayers after layer normalization. The input is fed into three different fully connected layers to generate the key (K), query (Q), and value (V) vectors in order to accomplish self-attention [21, 54, 55]. The basic attention equation is given by Eq. (1).

$$y_j = \sum_{i=1}^{N} W_{ij} x_i \tag{1}$$

where $W_{ij}x_i$ is the weighted sum of inputs x_i and weights W_{ij}. Keys, queries, and values are packed into the matrix K, Q, and V respectively. Each matrix will have a dimension of 18 × 768 ($d = 768$). K, Q, and V respectively are transformed with trainable matrix W^K, W^Q, W^V first. Input matrix X is stacked with transposed inputs x_i which can directly multiply with W^K, W^Q, W^V as shown in Eqs. (2–4).

$$XW^K = K \text{ and } K \in R^{N,S,D_K} \tag{2}$$

$$XW^Q = Q \text{ and } Q \in R^{N,S,D_K} \tag{3}$$

$$XW^V = V \text{ and } V \in R^{N,S,D_v} \tag{4}$$

where D_K represents dimensions of key and query tensor and D_V represents dimensions of value tensor.

To make weights interpretable and prevent output from becoming excessively large all weights $W_j = W_i j_{i=1}^N$ are squished using softmax, which gives probability values ranging from 0 to 1 as shown in Eqs. (5) and (7). Matrix product QK^T will measure the similarity among queries and keys. High probability is assigned to inputs with strong similarity and attention [54]. The scores are then downscaled by dividing them by the square root of the query's dimension, as indicated in Eq. (6). The key and the attention weights are multiplied by the value vector to get an output vector. Thus, the model learns more important and drop out the irrelevant words. Thus, the new attention is shown as Eq. (8):

$$Y = \sum_{i=1}^{N} W_{ij} V_i \tag{5}$$

$$W'_{ij} = \frac{Q \times K^T}{\sqrt{D_K}} (raw\, attention\, weights) \tag{6}$$

$$W_{ij} = softmax\left(W'_{ij}\right) \tag{7}$$

$$Y = softmax\left(\frac{Q \times K^T}{\sqrt{D_K}}\right) * V \tag{8}$$

Next, for a multi-headed attention computation, the key, query, and value undergo the self-attention process separately referred to as a head (h). Instead of single W^Q, W^K, W^V matrix we have h of each representing attention per query. The final output weight matrix i.e., $\alpha_{i,j}^{(h)}$ is computed using Eq. (9–11).

$$Q^{(h)}(x_i) = W_{h,q}^T x_i \tag{9}$$

$$K^{(h)}(x_i) = W_{h,k}^T x_i \tag{10}$$

$$V^{(h)}(x_i) = W_{h,v}^T x_i \tag{11}$$

After the multiplication of the value matrix to the product of query and key metrics, a softmax function is applied to get a more probabilistic output from the given tensor shown in the Eq. (12). Each head produces an output vector Y_j, that is combined into a single vector OM before proceeding to the final linear layer as shown in the Eqs. (13) and (14).

$$\alpha_{i,j}^{(h)} = \sigma_{soft\max}\left(\frac{\left\langle Q^{(h)}(x_i), K^{(h)}(x_i)\right\rangle}{\sqrt{d}}\right) \tag{12}$$

$$Y_j = \sum_{\forall j}^{h} \alpha_{i,j}^{(h)} V^{(h)}(x_i) \tag{13}$$

$$\mathrm{OM} = \mathrm{Concat}\,(h_1, h_2 \ldots..h_n) W_o \tag{14}$$

Finally, Layer normalisation is used as shown in Eq. (15), and the normalised residual output is propagated via a pointwise feed-forward network (FNN) composed of two linear layers separated by a ReLU activation [54]. The output is subsequently fed into the pointwise FNN and again normalised as shown in Eqs. (16–19). In this paper, the classification only involves a word embedding which is achieved using the uncased encoder layers. Thus, there is no requirement of decoder layers in our use case.

$$OM = Layer\, Norm(OM + x_i) \tag{15}$$

$$FNN_1 = W_{FNN_1}[OM] \tag{16}$$

$$FNN_1 = Relu(FNN_1) \tag{17}$$

$$FNN_2 = W_{FNN_2}[OM] \tag{18}$$

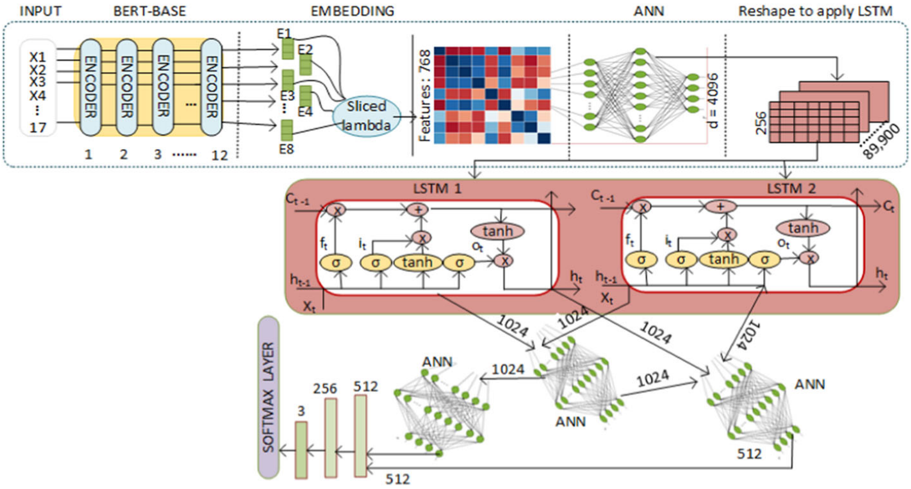


Fig. 8 Overview of RETN architecture

$$X_{encoder} = FNN_2 = LayerNorm(FNN_2 + OM) \tag{19}$$

4.3.2 The Embedding Batches

The proposed hybrid RETN model employs BERT-base which is based on the 8 self-attention multi-headed encoder network used for creating the word embeddings as shown in Fig. 8. The BERT-base contains 12 encoders which are stacked in the form of a multi-dimensional array. However, for the classification task, we only require the global summary vector thus we slice the array into 768 linear vectors using the SlicingOpLambda layer as shown in the detailed RETN architecture in Fig. 9.

Following that, the ANN layer is utilized, which employs the Rectified Linear Unit (ReLu) activation function and the L2 vector normalization with a weight regularization parameter of 0.001. The use of an ANN network improves performance and reduces overfitting in the model. Next, the feature vectors are computed and a reshape layer is created which batches the 4096 data variables to 16 batched 256 vectors. Before sending the array to the LSTM network, it is reshaped to produce a 2D matrix for LSTM. Moreover, the joint connections between LSTM and dense layers are used to form the residual connections. Moreover, an add or concatenate layer is used along with the residual network for joining the residual connections. Thus, the architecture is stacked in a specific manner to extract larger context by using LSTM and ANN networks concatenated into one long vector that is passed to the output layer used to make a prediction.

4.3.3 The LSTM Network

The LSTM [56] network is used because of its capability to capture a long-term correlation in sentences with arbitrary length, which is an added benefit to the self-attention. The LSTM's

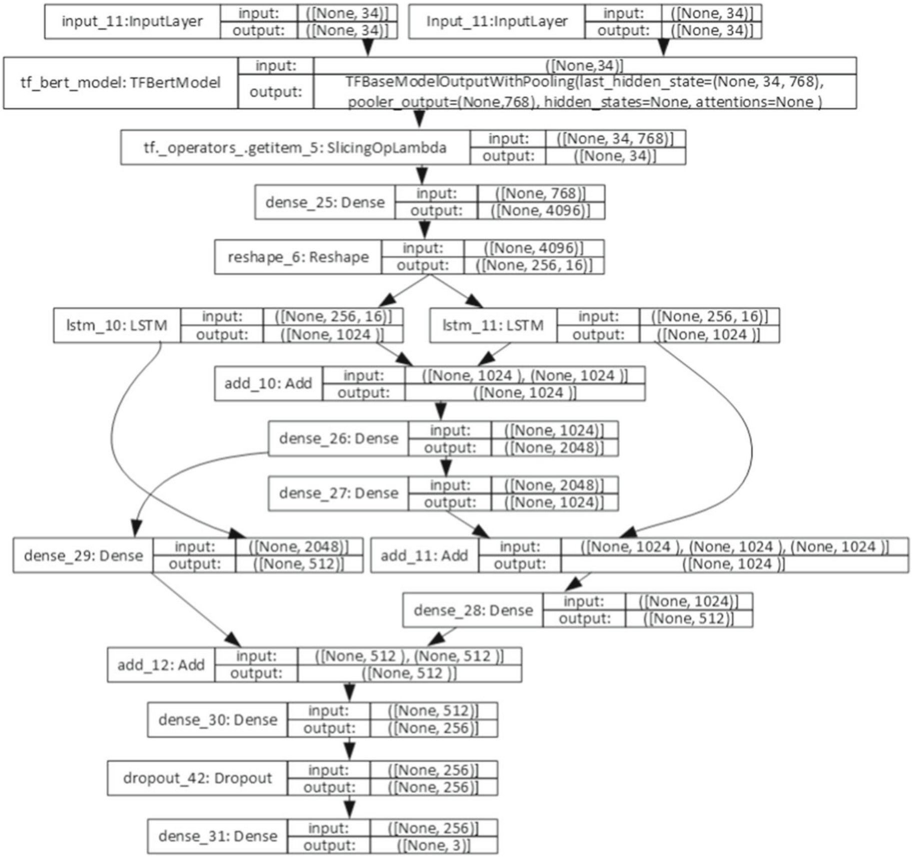


Fig. 9 Detailed RETN architecture

architecture allows it to overcome the "exploding gradient" and "vanishing gradient" problem using various gates namely input, forget, and output gates in addition to a cell state. The cell state works as long-term memory unit and acts as an additive connection between the states. The input x_t along with the intermediate state h_t at a given time t is provided to the LSTM cell and its output state is calculated by the following equations:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{20}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{21}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{22}$$

$$g_t = \tanh\left(W_g x_t + U_g h_{t-1} + b_g\right) \tag{23}$$

$$c_t = f_t o c_{t-1} + i_t o g_t \tag{24}$$

$$h_t = o_t o \tanh(c_t) \tag{25}$$

where W, U, and b represent the learnable parameters, sigmoid function is denoted by σ, and the convolution operator is denoted by o. The LSTMs gates are represented by the symbols f_t, i_t, and o_t, which stand for the forget, input, and output gates respectively. Further, the cell state or memory state is represented by c_t. The presence of cell states in LSTMs enable it to identify the long-term relationships present in the sentence. Therefore, it can be efficiently employed to the sentences with longer sequences.

In this architecture, the embedded batches are routed to two LSTM networks to capture sequential features and long-term contextual dependencies of a sentence. The outcome of multi-LSTMs is combined and fed to various peripheral divisions i.e., artificial neural networks (ANNs) with dense neural networks, creating the residual connections. This signifies that the feature set of the two LSTM networks is regenerated and the output of the ANN is extracted recursively. As a result, ANN networks produce a deep representation of the original sentence by extracting deep local features. In this paper, the proposed technique not only relies on attention training obtained from the transformers but also captures a global and local feature set using LSTMs and ANN networks. The pseudocode for our proposed work is shown in Algorithm 1.

ALGORITHM 1 RETN.

```
Input: $\boldsymbol{x}$ (a sequence of token IDs)
Output: $\boldsymbol{E}$, class prediction
Hyperparameters: $l_{max}$ (max sequence length); optimizer; $L$ (no' of encoder layers); loss; $H$ (no' of attention heads); $d_e$ (embedding of a token); $d_{mlp}$ (MLP dimensions); activation.
FUNCTION BERT-EMBED($\boldsymbol{x}$)
Parameters: "$\boldsymbol{We}$, $\boldsymbol{Wp}$" (the token and positional embedding matrices); $l$ (layer); $W_l$ (multi-head attention parameters for layer $l$); "$\gamma_l^1$, $\beta_l^1$, $\gamma_l^2$, $\beta_l^2$" (two sets of layer-norm parameters); "$W_{mlp1}^l$ , $b_{mlp1}^l$, $W_{mlp2}^l$, $b_{mlp2}^l$" (MLP parameters); "$W_f$, $b_f$, $\gamma$, $\beta$" (the final linear projection and layer-norm parameters); $W_u$ (unembedding matrix)
        $l \leftarrow$ length($\boldsymbol{x}$)
        for $t \in [l]$ : $e_t \leftarrow W_e$ [:, $x$ [$t$]] + $W_p$ [:, $t$]

        $\boldsymbol{X} \leftarrow [e_1, e_2, \ldots e_l]$
        for $l = 1, 2, \ldots, L$ do
                $\boldsymbol{X} \leftarrow \boldsymbol{X}$ + MultiHeadAttention($\boldsymbol{X}$ |$W_l$, Mask ≡ 1)
                for $t \in [l]$ : $\boldsymbol{X}$[:, $t$] ← layer_norm($\boldsymbol{X}$[:, $t$] | $\gamma_l^1$, $\beta_l^1$ )
                $\boldsymbol{X} \leftarrow \boldsymbol{X}$ + $W_{mlp2}^l$ RELU($W_{mlp1}^l \boldsymbol{X}$ + $b_{mlp1}^l 1$ | ) + $b_{mlp2}^l 1$
                for $t \in [l]$ : $\boldsymbol{X}$[:, $t$] ← layer_norm($\boldsymbol{X}$[:, $t$] | $\gamma_l^2$, $\beta_l^2$)
        end
        $\boldsymbol{X} \leftarrow$ RELU($W_f \boldsymbol{X}$ + $b_f$ 1)
        for $t \in [l]$ : $\boldsymbol{X}$[:, $t$] ← layer_norm($\boldsymbol{X}$[:, $t$] | $\boldsymbol{\gamma}$, $\boldsymbol{\beta}$)
        $\boldsymbol{E}$= softmax($W_u$ $\boldsymbol{X}$)
        return $\boldsymbol{E}$
END FUNCTION

FUNCTION RETN(x):
        weights ← Define weights
        Biases ← Define biases
        X ← dataset [FEATURES]. values
        Y ← dataset [CLASSES]. values
        Classes ← ['positive', 'negative', 'neutral']
        TRAIN_DATA, TEST_DATA, VALID_DATA ← TEST_TRAIN_SPLIT (final_dataset, 0.80, 0.20)
        INPUT ← INPUT (Input_length, type)
        BERT_EMBEDDINGS ← E
        DENSE ← DENSE_LAYER (BERT_Output_length, activation = 'ReLU')
        x ← RESHAPE(x)
        x ← DIMENSIONALITY_REDUCTION(x)
        LSTM_MODEL ← SEQUENTIAL_MODEL ({
                        LSTM_LAYER_1 (Output_length, activation = 'ReLU')
                        LSTM_LAYER_2 (Output_length, activation = 'ReLU')
                        })
        CONCATENATION ← (ADD (LSTM_LAYER_1, LSTM_LAYER_2), Output_length, activation = 'ReLU'))
        DENSE_1 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        DENSE_2 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        CONCATENATION ← (ADD (LSTM_LAYER_1, LSTM_LAYER_2, DENSE_2), Output_length,
        activation = 'ReLU'))
        DENSE_3 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        DENSE_4 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        CONCATENATION ← (ADD (DENSE_3, DENSE_4), Output_length, activation = 'ReLU')
        DENSE_5 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        DROP_OUT (0.5)
        DENSE_6 ← DENSE_LAYER (Output_length, activation = 'ReLU')
        PREDICTION ← OUTPUT (classes, activation = 'softmax')
        Loss ← categorial_cross-entropy, Optimizer ←Adam
        MODEL.COMPILE(Optimizer, Loss)
        MODEL_TRAIN (TRAIN_DATA, EPOCHS, VALID_DATA, BATCH_SIZE)
        return output
END FUNCTION
```

5 Experimental Results and Discussion

The experimentation is carried out using Anaconda Navigator with Python version 3.7.2 on Intel Core i5-83500H, an NVIDIA GeForce GTX 1050 GPU, and CPU of 2.40 GHz with 16 GB RAM. All of the ML and transfer models implemented in this paper are trained, tested, and validated using four COVID-19-related datasets. The training and testing samples are split into 80:20 ratio. A validation sample of 20% is drawn from the training dataset to avoid overfitting.

In this paper, GAN augmentation is used, which consists of CGAN to address the issue of multiclass imbalance as well as augmentation techniques such as random insertion, random swap, random deletion, and synonym replacement. The tweets present in COVIDSenti datasets highly suffered from a class imbalance in three sentiment categories i.e., negative, positive, and neutral. Using CGAN, the tweets are transformed into a more balanced form which are then used as an augmented training set to enhance the performance of classifiers used in COVID-19 sentiment classification. The training set data distribution and the data distribution obtained after applying the GAN in order to deal with the imbalance dataset problem are shown in Fig. 10a and b. The raw data distribution for positive, neutral, and negative tweets before applying GAN augmentation for the COVIDSenti dataset was 6280, 67,835, and 16,335, respectively which was then altered to 32,648, 45,000, and 42,464, respectively after the application of the same. Further, the graphs depicting the density distribution of the character length of a tweet for the COVIDSenti dataset are shown in Fig. 11a and b. This density distribution shows that before GAN augmentation the character length of a tweet was relatively less, whereas post-GAN augmentation tweets shown a gradual increase in distribution of the character length of tweets. However, the normal distribution curve is observed in both the graphs depicting that the text lengths follow the normalized form.

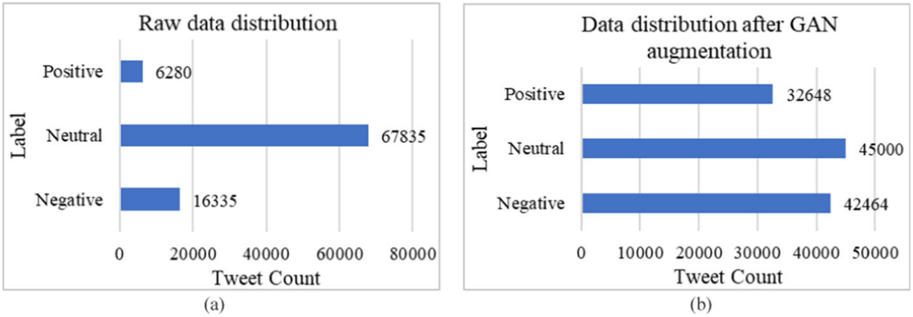


Fig. 10 Data distribution for COVIDSenti dataset a Before class balancing, and **b** After class balancing using CGAN

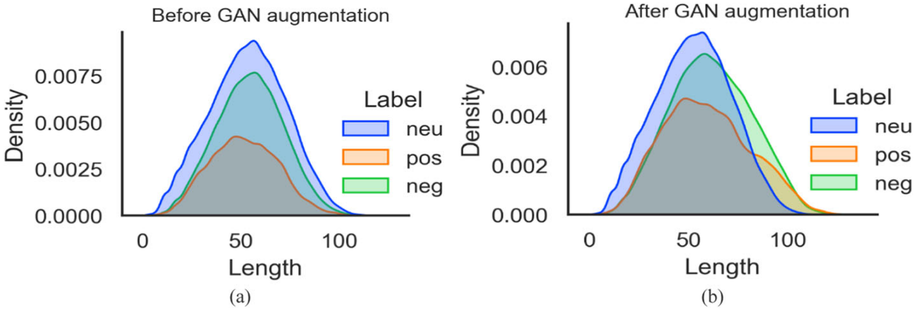


Fig. 11 The sentence length distributions for COVIDSenti dataset **a** Before class balancing, and **b** After class balancing using CGAN

The proposed RETN model employs different loss functions and optimizers on the COVID-related Twitter data. Optimizations such as stochastic gradient descent (SGD), mini-batch stochastic gradient descent, and Adaptive Gradient (Adagrad) allow it to reach global minima but are inefficient enough to improve the architecture's weights and biases. It has been observed from the results that the Root Mean Square Propagation (RMSProp) loss and the Adaptive Moment Estimation (Adam) performed equally well. The Adam optimizer with a learning rate of 3e−5 and a maximum sequence length of 1024 is used in this study. Further, the proposed RETN model is trained in 64 batch sizes and the dropout rate is set to 0.1 for dense layers. Table 4 shows the parameter setting for the proposed model.

To calculate the performance of ML and transfer learning models on four COVIDSenti datasets various performance metrics are utilized. The performance metrics namely, precision, F1-score, recall, and accuracy are utilized to evaluate and compare the performance of the proposed RETN hybrid model with other state-of-the-art techniques. A confusion matrix [57] is used to summarize the performance of the proposed model using multiple sub-metrics

Table 4 Parameters used in the proposed hybrid RETN model

Layer (type)	Output shape	Parameters	Connected to
input_11 (input layer)	[(None, 34)]	0	–
input_12 (input layer)	[(None, 34)]	0	–
tf_bert_model (TFBertModel)	TFBaseModelOutput	109,482,240	input_11 [0] [0] input_12 [0] [0]
tf.operators.getitem_5	(None, 768)	0	tf_bert_model [5] [0]
dense_25 (dense)	(None, 4096)	3,149,824	tf.__operators__.getitem_5 [0] [0]
reshape_6 (reshape)	(None, 256, 16)	0	dense_25 [0] [0]
lstm_11 (LSTM)	(None, 1024)	4,263,936	reshape_6 [0] [0]
lstm_10 (LSTM)	(None, 1024)	4,263,936	reshape_6 [0] [0]
add_10 (add)	(None, 1024)	0	lstm_10 [0] [0] lstm_11 [0] [0]
dense_26 (dense)	(None, 2048)	2,099,200	add_10 [0] [0]
dense_27 (dense)	(None, 1024)	2,098,176	dense_26 [0] [0]
add_11 (add)	(None, 1024)	0	lstm_11 [0] [0] lstm_10 [0] [0] dense_27 [0] [0]
dense_28 (dense)	(None, 512)	524,800	add_11 [0] [0]
dense_29 (dense)	(None, 512)	1,049,088	dense_26 [0] [0]
add_12 (add)	(None, 512)	0	dense_28 [0] [0] dense_29 [0] [0]
dense_29 (dense)	(None, 256)	131,328	add_12 [0] [0]
dropout_42 (dropout)	(None, 512)	0	dense_30 [0] [0]
dense_31 (dense)	(None, 3)	771	dropout_42 [0] [0]
Total parameters		127,063,299	
Trainable parameters		127,063,299	
Non-trainable parameters		0	

[58], i.e., True Negative, False Positive, True Positive, and False Negative. The performance of each category in a classification problem is analyzed separately. These performance metrics have been specifically selected to demonstrate the model's capability to achieve the best classification results.

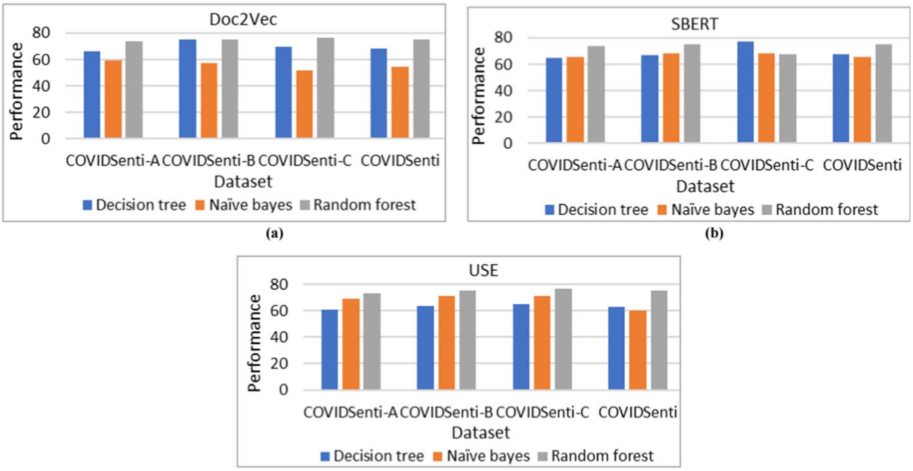
5.1 Comparative Analysis

In this study, various experiments are undertaken to compute the classification results using the four COVID-19-related Twitter datasets. Various traditional ML algorithms such as NB, DT, and RF are implemented and the results are compared with transfer learning models and the proposed hybrid RETN model. To utilize conventional ML models, a one-dimensional distribution of all word vectors is created. This is accomplished by employing feature extraction techniques such as Doc2Vector, SBERT, and USE. Table 5 shows the accuracy of various ML algorithms obtained using various word embedding techniques.

In context of the Doc2Vec word embedding technique on the COVIDSenti dataset, the RF approach achieves the highest accuracy of 75.42%, whereas other ML methods, such as DT and NB, attain the accuracies of 68.24%, and 54.50%, respectively. However, the DT approach achieves the highest accuracy of 75.08% compared to NB and RF models on the COVIDSenti-B dataset. While using the SBERT word embedding technique, the RF model surpasses other ML methods, such as DT and NB with the best accuracy of 75.41% on COVIDSenti and COVIDSenti-B datasets. However, the DT model attains the best accuracy of 76.81% as compared to NB and RF models on the COVIDSenti-C dataset. Additionally, the word embedding technique i.e., USE is used with different ML classifiers, resulting in highest accuracy of 76.48% on the COVIDSenti-C dataset for the RF classifier. Finally, it is concluded from the results obtained that the RF model performed exceptionally well with Doc2Vec and USE word embeddings on all four COVID-19 datasets. The graphical representation showing comparative analysis of the RF, DT, and NB classifiers using various word embeddings on the four COVID-19-related datasets is shown in Fig. 12a–c. It is observed that the highest accuracy tends to be around 70–77% across all four datasets using RF classifier.

Table 5 Accuracy comparison of machine learning classifiers with different word embeddings

Models\Datasets	COVIDSenti	COVIDSenti-A	COVIDSenti-B	COVIDSenti-C
Doc2Vec				
DT	0.68240	0.66120	0.75080	0.69850
NB	0.5450	0.5950	0.5740	0.5200
RF	0.75420	0.73560	0.74890	0.76450
SBERT				
DT	0.67480	0.64850	0.66890	0.76812
NB	0.65540	0.65800	0.68530	0.68240
RF	0.75410	0.73500	0.75410	0.67240
USE				
DT	0.63150	0.61000	0.63520	0.65230
NB	0.60230	0.69230	0.71450	0.71410
RF	0.75410	0.73120	0.75410	0.76480



(a)

(b)

(c)

Fig. 12 Comparison of machine learning algorithms using different word embeddings: **a** Doc2Vec, **b** SBERT, and **c** USE

After ML word embedding, various transfer learning techniques are utilized to achieve better sentiment classification performance. While working with transfer learning techniques, we experimented with three transfer learning models, namely, BERT, DistilBERT, and RoBERTa. The results obtained from the transfer learning models are also compared to the proposed hybrid RETN model. The comparative analysis of results for various transfer learning models is shown in Table 6. In the case of the data pre-processing using GAN augmentation, the BERT and DistilBERT achieved the best accuracies of 96% and 96.9% on the COVIDSenti-A dataset. The BERT architecture achieves the maximum precision, F1-score, and recall, i.e., 96.7%, 93%, and 91.4%, respectively. However, on the same dataset, the DistilBERT architecture perceives the maximum precision, recall, and F1-score, i.e., 96.66%, 91%, and 93.3%, respectively. Furthermore, BERT and DistilBERT show comparable performances on the COVIDSenti-C and COVIDSenti-B datasets for all the performance metrics. However, BERT and DistilBERT attain a low accuracy of 88% and 75% on the COVIDSenti dataset compared to the proposed RETN model. However, DistilBERT shows high precision, recall, and F1-score, i.e., 95%, 91%, and 92%, respectively, as compared to BERT on the COVIDSenti dataset. The RoBERTa observes the low performance for all the evaluation metrics on all the four COVIDSenti datasets. It is observed that BERT and DistilBERT show comparable performance in terms of accuracy, i.e., 98%, to the proposed RETN model on the COVIDSenti-C dataset. However, they show slightly high performance on COVIDSenti-A with an accuracy of 96%, and 96.9% and for COVIDSenti-B with an accuracy of 96% and 98.5% respectively. Finally, the proposed RETN model attained an accuracy of 95% for the COVIDSenti-A, 97% for COVIDSenti-B, and 98% for COVIDSenti-C datasets. Moreover, the proposed RETN model achieves the highest accuracy, precision, recall, and F1-score, i.e., 98%, 95%, 97%, and 96% on the COVIDSenti dataset, as compared to other transfer learning models.

From Table 6, it can be observed that the classification performance with GAN augmentation achieved better results than without augmentation. Thus, employing GAN augmentation

Table 6 Comparison of transfer learning models on test data

Data	Models\datasets	COVIDSenti-A				COVIDSenti-B				COVIDSenti-C				COVIDSenti			
		Acc	Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1	Acc	Pre	Rec	F1
Using GAN augmentation	BERT	0.960	0.967	0.914	0.93	0.960	0.97	0.96	0.963	0.98	0.96	0.98	0.97	0.88	0.83	0.73	0.77
	DistilBERT	0.969	0.966	0.91	0.933	0.985	0.97	0.96	0.96	0.98	0.98	0.96	0.97	0.75	0.95	0.91	0.92
	RoBERTa	0.900	0.866	0.82	0.84	0.750	0.58	0.33	0.29	0.770	0.26	0.33	0.29	0.757	0.25	0.33	0.29
	RETN	**0.95**	**0.91**	**0.893**	**0.90**	**0.97**	**0.943**	**0.96**	**0.953**	**0.98**	**0.95**	**0.963**	**0.96**	**0.98**	**0.95**	**0.97**	**0.96**
Without GAN augmentation	BERT	0.94	0.92	0.88	0.90	0.93	0.90	0.88	0.89	0.93	0.89	0.87	0.88	0.93	0.88	0.88	0.88
	DistilBERT	0.94	0.91	0.88	0.90	0.93	0.87	0.87	0.87	0.90	0.87	0.86	0.86	0.92	0.86	0.88	0.87
	RoBERTa	0.89	0.83	0.75	0.78	0.89	0.85	0.76	0.80	0.89	0.83	0.78	0.80	0.89	0.83	0.80	0.81
	RETN	**0.94**	**0.91**	**0.88**	**0.89**	**0.94**	**0.91**	**0.89**	**0.90**	**0.93**	**0.88**	**0.86**	**0.879**	**0.93**	**0.88**	**0.87**	**0.87**

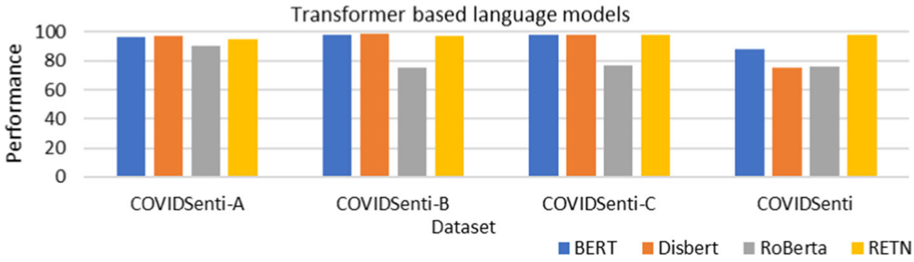
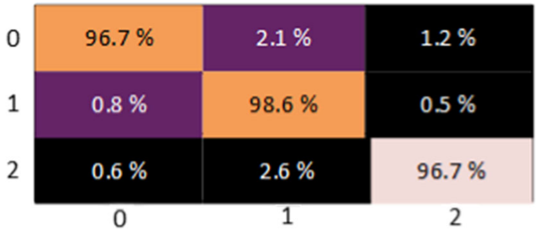


Fig. 13 Accuracy comparison of transformer-based language models with the hybrid RETN model



	0	1	2
0	96.7 %	2.1 %	1.2 %
1	0.8 %	98.6 %	0.5 %
2	0.6 %	2.6 %	96.7 %

Fig. 14 Confusion matrix for RETN model

improves the proposed model's accuracy by 1% for the COVIDSenti-A, 3% for COVIDSenti-B, 5% for COVIDSenti-C, and 5% for COVIDSenti datasets. This indicates that the GAN augmentation is generally helpful and effective. In Fig. 13, the graphical representation of comparative analysis of different transformer-based architectures namely, BERT, DistilBERT, and RoBERTa illustrates that accuracies are ranged between 90 and 98%. As a result, it is verified from the classification outcomes that ML models are less capable than pre-trained transfer learning models in extracting the efficient feature embedding required for classification. Although, transfer learning models namely, BERT, DistilBERT, and RoBERTa performed better than conventional ML models but fail to perform better than the proposed hybrid RETN model on all the four COVIDSenti datasets.

In Fig. 14, the confusion matrix is used as a metric to evaluate the validity of the classification performance of the proposed RETN model on the COVIDSenti dataset. In the case of tweets with positive sentiments, 96.7% are accurately identified as positive, while 2.1% and 1.2% of tweets are incorrectly identified as negative and neutral. Further, 98.6% of all tweets with negative sentiments are correctly identified as negative, while 0.5% and 0.8% of tweets are incorrectly identified as neutral and positive. Moreover, 96.7% of all tweets with neutral sentiments are correctly identified as neutral, whereas 0.6% and 2.6% of tweets are incorrectly identified as negative and positive.

All the transfer learning models utilized in this study, i.e., BERT, DistilBERT, RoBERTa, and the proposed RETN model are trained and tested on the training and testing datasets. However, the performance evaluation is performed on the validation dataset. The proposed hybrid RETN model performs well on the training data with the accuracy ranging between 95 and 98.5% on all the four COVIDSenti datasets as shown in Table 7. Further, it shows comparable performance on validation and testing datasets with the accuracy ranging between 94 and 98% on all the four COVIDSenti datasets. In addition, Fig. 15 depicts graphical representation of the experimental results obtained using various transfer learning techniques on testing, training, and validation datasets for four COVIDSenti datasets. Therefore, the RETN

Table 7 Accuracy comparison of transformer-based language models on training and validation data

Datasets\Models	BERT	DistilBERT	RoBERTa	RETN
COVIDSenti				
Testing	0.88	0.75	0.757	0.98
Training	0.937	0.944	0.743	0.975
Validation	0.937	0.952	0.753	0.950
COVIDSenti-A				
Testing	0.96	0.969	0.90	0.95
Training	0.945	0.949	0.875	0.952
Validation	0.829	0.849	0.847	0.949
COVIDSenti-B				
Testing	0.96	0.985	0.75	0.97
Training	0.944	0.955	0.847	0.956
Validation	0.863	0.850	0.743	0.953
COVIDSenti-C				
Testing	0.98	0.98	0.77	0.981
Training	0.970	0.969	0.764	0.949
Validation	0.87	0.869	0.762	0.946

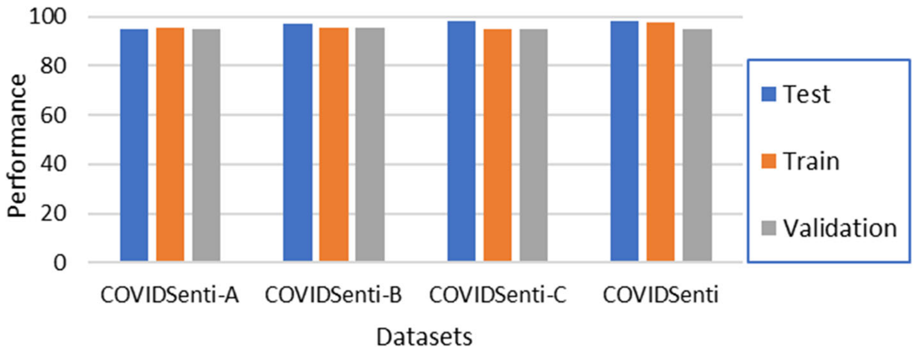


Fig. 15 Performance comparison of RETN model using test, train and validation accuracy for four COVIDSenti datasets

model is good at classifying the tweets without overfitting the data, thereby making accurate classifications for the data it has not seen before. It is because of the key characteristics of the proposed hybrid model RETN results from the transformer-based word embeddings and the residual bilingual connection between the layers. Moreover, the proposed RETN model ensures that the weights and biases do not override each other causing a “vanishing gradient” or “exploding gradient” problem [59]. On the other hand, other transfer learning models i.e., BERT, DistilBERT, and RoBERTa perform relatively well in the training datasets but show poor performance in the validation and testing datasets. Thus, BERT, DistilBERT, and RoBERTa have overfitting issues and comparatively low performance as compared to the proposed RETN model.

Next, the training and validation dataset metrics are used, and the plots are presented over each epoch to verify the effectiveness of the proposed hybrid RETN model. The accuracy and loss plots for validation and training datasets for all the four COVIDSenti datasets are shown in Fig. 16a–d. The accuracy and loss curves illustrate the decrease of loss and increase in accuracy concerning the number of epochs. The huge gap between the validation and training accuracy indicates the overfitting situation. However, in our case the validation and training accuracy (or loss) curves are close to each other, thereby, avoiding the overfitting.

5.2 Comparative Analysis with Baseline Approach

To validate the efficacy of the proposed hybrid RETN model, we compared our results to those of the state-of-the-art study [15] which used identical COVIDSenti datasets as employed in our study. The evaluation metric i.e., accuracy has been utilized in the state-of-the-art literature to evaluate the performance of the various classification models.

In comparison to the state-of-the-art models, the proposed methodology uses recall, precision, and F1-score in addition to accuracy metric. The comparative results are graphically shown in Fig. 17. The state-of-the-art study fine-tuned the transfer learning models such as BERT, XLNET, ALBERT, and DistilBERT. Moreover, the state-of-the-art study proposed two hybrid models, namely the Hybrid Ranking Model (HyRank) and the IWV model. Furthermore, among these baseline models, BERT achieved the highest accuracy of 94.8, 94.1, 93.7, and 93.2 on COVIDSenti, COVIDSenti-A, COVIDSenti-B, and COVIDSenti-C, respectively. In this paper, a hybrid transfer learning and DL-based RETN model is proposed that attained the highest accuracy as compared to all the state-of-the-art approaches. Experimental outcomes verify that the proposed hybrid RETN model attains the best accuracy of 95% with 0.9% gain on COVIDSenti, 97% with 3.3% gain on COVIDSenti-A, 98.1% with 4.9% gain on COVIDSenti-B, and 98% with 3.2% gain on COVIDSenti-C, respectively.

It is concluded that the proposed hybrid RETN model is a language summarization network that can attract both global and local attention while avoiding the "exploding gradient" and "vanishing gradient" problem. It is achieved by integrating the transformer-based language model i.e., the pre-trained BERT model and DL-based multi-LSTM networks. Traditional RNN or LSTM-based models evaluate the text input sequence word by word [10]. In contrast, the pre-trained BERT model does not evaluate the text input sequence word by word, but takes the full sequence as input all at once and generates word embeddings. LSTM, on the other hand, extracts important contextual and sequential information, while avoiding the "exploding gradient" and "vanishing gradient" problem [11]. Moreover, the outcome from multi-LSTMs is again fed to various residual connections between layers to improve the performance by further extracting the features in a residual manner rather than the traditional sequential manner. This is due to the residual connection producing a non-denominational gradient and the normal updation of weights and biases. Thus, the proposed hybrid RETN model is efficient in both sequence modeling as well as extracting important contextual features.

5.3 Word Cloud

A word cloud [60] is a text visualization method for evaluating information presented in the text. It indicates the importance of words by showing the most frequently used words in bigger font in comparison to the least frequently used words. In this paper, we have plotted and compared the positive and negative word cloud for a huge COVIDSenti dataset to know

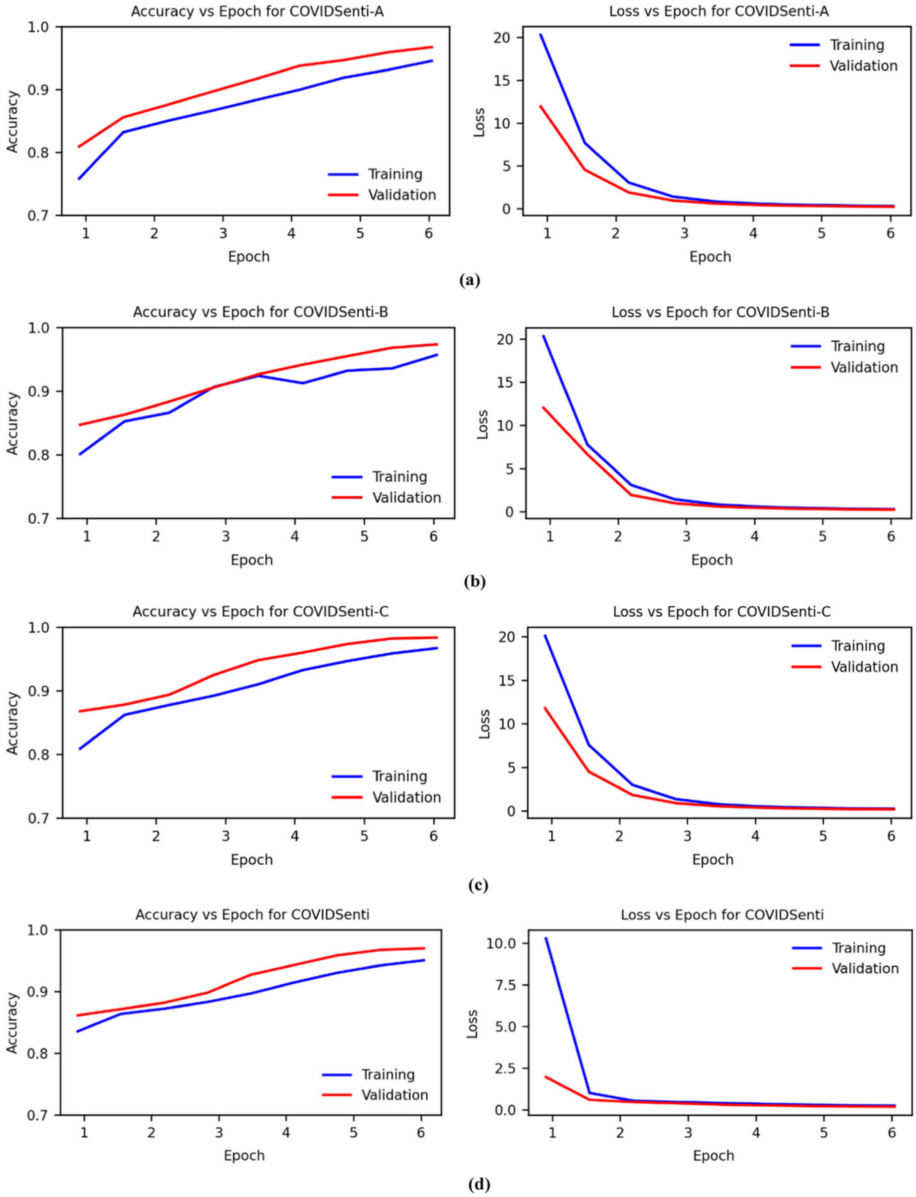
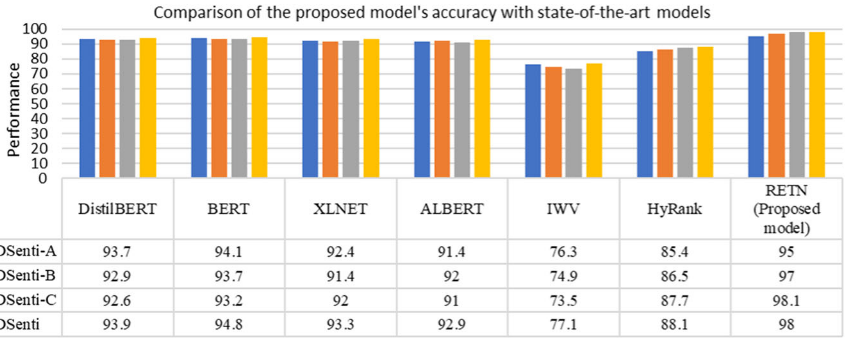


Fig. 16 Performance comparison of RETN model **a** Accuracy and loss graph for COVIDSenti-A; **b** Accuracy and loss graph for COVIDSenti-B; **c** Accuracy and loss graph for COVIDSenti-C; and **d** Accuracy and loss graph for COVIDSenti



	DistilBERT	BERT	XLNET	ALBERT	IWV	HyRank	RETN (Proposed model)
■ COVIDSenti-A	93.7	94.1	92.4	91.4	76.3	85.4	95
■ COVIDSenti-B	92.9	93.7	91.4	92	74.9	86.5	97
■ COVIDSenti-C	92.6	93.2	92	91	73.5	87.7	98.1
■ COVIDSenti	93.9	94.8	93.3	92.9	77.1	88.1	98

Fig. 17 Comparison of the RETN model with state-of-art models on same dataset

the people's opinions, attitude, sentiment towards COVID-19 pandemic as shown in Fig. 18a and b.

An analysis of the most commonly used words presented in the word cloud is given below:

- Words such as virus, coronavirus, outbreak, corona, covid, and others are frequently used in both positive and negative tweets, representing an individual's perspective and responses during the COVID-19 pandemic.
- The word cloud for negative tweets includes words like pandemic, death, worry, suicide, fear, infection, quarantined, die, deadly, spreading, epidemic, kill, worried, and so on, depicting the challenges that people have experienced during times of global crisis.
- The word cloud for positive tweets includes words like hope, getting doctor, health, protect, positive, emergency, sign, medical, and so on, signifying positivity during the outbreak.

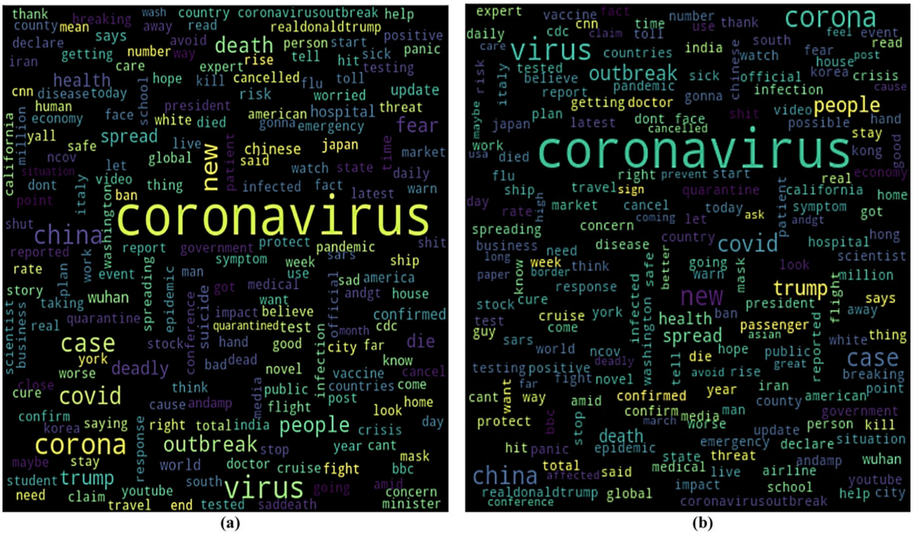


Fig. 18 **a** Word Cloud of negative tweets; **b** Word Cloud of positive tweets

- Both word clouds include words representing actions such as, avoid, stay safe, travel ban, fight back, etc., revealing how individuals expressed their concerns about the COVID-19 pandemic by posting their views on the Twitter platform.

It is concluded from plot analysis that people more actively reported their sentiments on social media platforms during the COVID-19 pandemic. As a result, it has been confirmed that social media data can be utilised for text classification to predict sentiments even during COVID-19 pandemic.

5.4 Statistical Test

In this paper, the Friedman rank test [61] is used to compare various classifiers on different datasets. This test is a nonparametric variant of the repeated measurements ANOVA test and is based on the classification algorithm's average ranking performance on each task. The statistic test determines the statistically significant difference between the classifiers. In this test, k algorithms are compared on N datasets with r_i denoting $i - th$ algorithm's average order value. r_i follows the normal distribution, with mean and variance of $(k + 1)/2$ and $(k2 - 1)/12$, respectively. The Friedman statistic with $k - 1$ degrees of freedom is expressed as Eq. (26). The more relevant statistic with $k - 1$ and $(k - 1)(N - 1)$ degrees of freedom is distributed according to the F-distribution [61]. This statistic is known as the Friedman (F), and it is represented as follows in Eq. (27):

$$\gamma_x^2 = \frac{12N}{k(k+1)}\left(\sum_{i=1}^{k} r_i - \frac{k(k+1)^2}{4}\right) \tag{26}$$

$$\gamma_F = \frac{N - 1\gamma_x^2}{N(k-1) - \gamma_x^2} \tag{27}$$

In this study, the Friedman test produces a p-value of 0.044, which is less than the significant level (α) of 0.05. This rejects the hypothesis that the performance of all models is similarly deducing that at least one of the models is unique. The Friedman test ranks the algorithms implemented on each dataset from best to worst in terms of performance. Table 8 shows the ranking of four COVID-19 datasets based on the Friedman test. The proposed RETN model obtains the best rank of 1, BERT algorithm ranks 2, and are followed by DistilBERT, RoBERTa, USE + RF, Doc2Vec + RF, SBERT + RF, Doc2Vec + DT, USE + NB, SBERT + DT, SBERT + NB, USE + DT and Doc2Vec + NB algorithms. The critical value of the F distribution $F(k - 1, (k - 1)(N - 1))$ is obtained as 8.47. The ties in this rank are solved by taking the average of their ranks. Thus, it is concluded that the RETN algorithm produces better outcomes than other algorithms.

If the Friedman test's null hypothesis is rejected, post-hoc tests can be performed to complement the statistical analysis. In this paper, the Nemenyi test is used to obtain the differences between the classifiers used, and the critical value (CD) is calculated. The Nemenyi test [61] compares all classifiers with each other. As a result, the performance of two classifiers differs considerably if their rankings differ by at least a critical difference. The following formula is used to determine the critical difference:

$$CD = q_{\propto}\sqrt{\frac{k(k+1)}{6N}} \tag{28}$$

The value of $q_{\propto} = 8.47$, and the critical value (CD) $= 9.12$ is calculated based on $q_{\propto}$. Figure 19 depicts the Friedman test chart based on the experimental results obtained. Accord-

Table 8 Ranking based on the Friedman test

Models versus rank	RETN	DistilBERT	BERT	RoBERTa	Doc2Vec + DT	Doc2Vec + NB	Doc2Vec + RF	SBERT + DT	SBERT + NB	SBERT + RF	USE + DT	USE + NB	USE + RF
Mean Rank	2.0	2.75	2.25	4.5	8.0	13.0	5.87	9.37	10.0	6.5	11.7	9.25	5.75
Final Rank	1	3	2	4	8	13	6	10	11	7	12	9	5

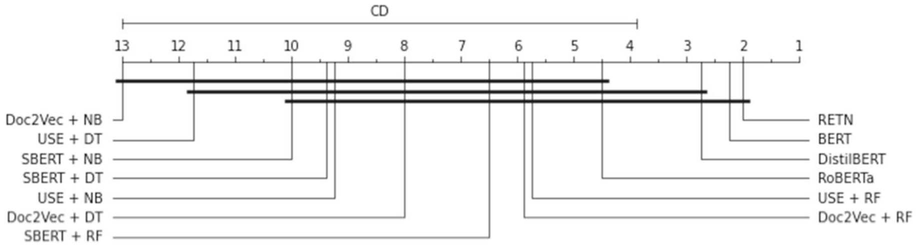


Fig. 19 Friedman test chart for various models

ing to the Friedman test rank, the Nemenyi test reveals that RETN, BERT, and DistilBERT algorithms are significantly distinct from RoBERTa, USE + RF, Doc2Vec + RF, SBERT + RF, Doc2Vec + DT, USE + NB, SBERT + DT, SBERT + NB, USE + DT, and Doc2Vec + RF algorithms. RETN outperforms both BERT and DistilBERT. Finally, the statistical tests show that, of all the classifiers examined in this work, the RETN and BERT perform the best among all datasets.

5.5 Benchmark Testing

The proposed hybrid RETN model's performance is evaluated using three benchmark datasets. Several ML-based classifiers used in text classification, such as RF, NB, and DT, as well as state-of-the-art architectures such as XLNET, BERT, ALBERT, and DistilBERT, are used to evaluate results. A brief overview of the benchmark datasets used in this study is given below:

(1) **Amazon review sentiment prediction dataset:** This dataset comprises few million Amazon customer reviews as input text and star ratings as output labels. It is divided into two classes: class X and class Y. Moreover, class X has a rating of one and two stars, whereas class Y has a rating of four and five stars. [62].

(2) **Suicide and depression detection dataset:** This dataset comprises of comments obtained from the Reddit platform's "SuicideWatch" and "depression" subreddits. All comments related to "SuicideWatch" were collected from December 16, 2008, to January 2, 2021, while "depression" comments were collected from January 1, 2009, to January 2, 2021 [63].

(3) **Sentinet140 dataset:** It includes 1,600,000 tweets and has already been labeled as 0 for negative, 2 for neutral, and 4 for positive [64].

The accuracy metric is used to evaluate the classification outcome of the proposed hybrid model. Moreover, K-fold cross-validation [65] with a value of K set to 5 is utilized to examine the performance of our model on different datasets. The classification outcomes for three benchmark datasets are presented in Tables 9, 10, and 11. It is found that our proposed method outperforms all other traditional ML and transfer learning methods. The proposed RETN model obtained accuracy of 99.26%, 99.66%, and 99.57% on sentinet140, suicide and depression and amazon review datasets, respectively.

Table 9 Sentinet140 dataset

Model	CV1	CV2	CV3	CV4	CV5	Mean
Random forest	0.5415	0.568	0.5345	0.564	0.564	0.55
Naïve Bayes	0.522	0.511	0.52	0.519	0.505	0.51
Decision tree	0.527	0.548	0.543	0.528	0.5395	0.54
BERT	0.9881	0.9943	0.9831	0.9925	0.9918	0.9820
XLNET	0.9793	0.9731	0.9693	0.9787	0.9725	0.9746
ALBERT	0.7381	0.7387	0.7331	0.7493	0.7562	0.7431
DistilBERT	0.9918	0.9887	0.9856	0.9862	0.9887	0.9882
Proposed model	0.9925	0.9925	0.9943	0.9912	0.9925	0.9926

Table 10 Suicide and depression dataset

Model	CV1	CV2	CV3	CV4	CV5	Mean
Random forest	0.762	0.7555	0.756	0.747	0.7495	0.75
Naïve Bayes	0.505	0.527	0.4860	0.5055	0.5055	0.53
Decision tree	0.6735	0.65	0.6725	0.6375	0.671	0.67
BERT	0.9375	0.9475	0.9500	0.9300	0.9575	0.9445
XLNET	0.9775	0.9700	0.9725	0.9850	0.9700	0.9750
ALBERT	0.9575	0.9775	0.9675	0.9500	0.9625	0.9630
DistilBERT	0.9425	0.9500	0.9450	0.9225	0.9475	0.9415
Proposed model	0.99750	0.9950	0.9956	0.9987	0.9962	0.9966

Table 11 Amazon review dataset

Model	CV1	CV2	CV3	CV4	CV5	Mean
Random forest	0.5731	0.5829	0.59	0.5808	0.5702	0.58
Naïve Bayes	0.4854	0.4860	0.4860	0.4856	0.4868	0.49
Decision tree	0.5579	0.5702	0.5622	0.5652	0.5591	0.56
BERT	0.8775	0.9225	0.9050	0.8925	0.8975	0.8990
XLNET	0.8925	0.9075	0.9000	0.9100	0.8900	0.9000
ALBERT	0.9250	0.9075	0.8925	0.9100	0.9175	0.9105
DistilBERT	0.8650	0.9150	0.8900	0.8875	0.8875	0.8890
Proposed model	0.9956	0.9950	0.9937	0.9968	0.9975	0.9957

6 Conclusion

Sentiment analysis is a powerful tool for determining public attitudes in order to take effective health measures during a pandemic. Various challenges that come forth as a result of the COVID-19 pandemic necessitate close monitoring of how people perceive the growing threat and respond to guidelines and policies on the social platforms. The goal of this research is to develop a transformer-based hybrid model to perform sentiment analysis on Twitter data by combining neural networks and transfer learning. CGAN was used as a data balancing technique, and various data augmentation techniques such as random swapping, random deletion, and random insertion are used to create a dataset to train on. Moreover, statistical analysis is performed to validate the performance attained by the various models employed in our study, showing that the RETN model has adequate features to improve sentiment classification performance. The answers to the formulated research questions are concluded as follows:

1. The proposed RETN hybrid model efficiently obtains the remarkable performance for four largescale benchmark COVIDSenti datasets i.e., COVIDSenti, COVIDSenti-A, COVIDSenti-B, and COVIDSenti-C. The proposed model extracts the relevant feature set, improves accuracy, and overcomes the limitations of traditional machine learning and deep learning models. RETN not only outperforms traditional ML techniques implemented with various word embeddings, but also outperforms various transformer-based models. As a result, the chosen hybrid RETN language model is effective for sentiment analysis.
2. The transfer learning incorporated in the BERT architecture has effectively aided the model in learning text embeddings while preserving the syntactic and semantic meanings of the context and giving more attention to important words for analyzing online attitudes of users on the Twitter platform during the COVID-19 pandemic. The proposed hybrid model RETN is made up of BERT-base that has been fine-tuned and trained using labeled dataset. The extracted BERT embeddings are combined with the LSTMs and then fed to the dense layers. This optimizes the model by addressing the problems of "vanishing gradient" and "exploding gradient", lowering the risk of becoming stuck in local minima and preventing the problems of overfitting and underfitting.
3. The experimental results achieved by using the proposed hybrid model on four COVIDSenti datasets show that it outperforms RF, DT, NB, BERT, RoBERTa, and DistilBERT. We achieved a remarkable accuracy of 95% on the COVIDSenti-A dataset, 97% on the COVIDSenti-B dataset, 98% on the COVIDSenti-C dataset, and 98% on the COVIDSenti dataset. Finally, benchmark testing performed on three real-world text classification datasets yields promising results for the proposed hybrid model. As a result, it is concluded that RETN's language modeling capability contributes significantly to a good text classification model.

In the future work, we aim to improve the performance by using more advanced hybrid models such as BERT-GRU, BERT-biLSTM in combination with the nature inspired optimization techniques. It may be worth experimenting with larger transformer architectures in the future, such as the BERT-Large, GPT-2, and GPT-3 architectures. We also intend to extend the experimentation by considering real-time data and obtaining efficient models by determining the optimal hyperparameters. More data in the form of text, images, or audio can be considered for research in the future. Furthermore, taking into account the requirement of transformer networks to fully generalize different features based on comparison with or without augmentation can be addressed in the future. As per the current transformer research trends, it is reasonable to investigate and develop various hybrid word embedding techniques in the future. This could certainly aid in the use of large datasets of one or two million tweets in various languages, including hashtags, as well as image and image captioning systems, allowing for the development of a more appropriate COVID-19 sentiment analysis model.

Data Availability The Twitter data related to COVID-19 pandemic used in this study is available at Usman Naseem et al. [15].

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Guo YR, Cao QD, Hong ZS, Tan YY, Chen SD, Jin HJ, Yan Y (2020) The origin, transmission and clinical therapies on coronavirus disease 2019 (COVID-19) outbreak–an update on the status. Mil Med Res 7(1):1–10
2. Worldometers (2022) coronavirus death cases 2021. [Online]. https://www.worldometers.info/coronavirus/. (Accessed: 24 Jan 2022)
3. Merchant RM, Lurie N (2020) Social media and emergency preparedness in response to novel coronavirus. JAMA 323(20):2011–2012
4. Kutana S, Lau PH (2021) The impact of the 2019 coronavirus disease (COVID-19) pandemic on sleep health. Can Psychol 62(1):12
5. Appel G, Grewal L, Hadi R, Stephen AT (2020) The future of social media in marketing. J Acad Market Sci 48:79–95
6. Gadde V, Derella M (2020) An update on our continuity strategy during COVID-19. Twitter. 2020. [Online]. https://blog.twitter.com/en_us/topics/company/2020/covid-19 (Accessed: 20 Jan 2022)
7. Internet Users Worldwide Statistic, [Online]. https://www.broadbandsearch.net/blog/internet-statistics,Anonymous, (Accessed: 22 Jan 2022)
8. Choi S, Lee J, Kang M-G, Min H, Chang Y-S, Yoon SJM (2017) Large-scale machine learning of media outlets for understanding public reactions to nation-wide viral infection outbreaks. Methods 129:50–59
9. Saeed Z, Abbasi RA, Razzak I, Maqbool O, Sadaf A, Xu G (2019) Enhanced heartbeat graph for emerging event detection on Twitter using time series networks. Expert Syst Appl 136:115–132
10. Culurciello E (2018) The fall of RNN/LSTM—towards data science. [Online]. https://towardsdatascience.com/the-fall-of-rnn-lstm-2d1594c74ce0 (Accessed: 5 Jan 2022)
11. Liu R, Shi Y, Ji C, Jia M (2019) A survey of sentiment analysis based on transfer learning. IEEE Access 7:85401–85412
12. Chowdhury AA, Das A, Saha SK, Rahman M, Hasan KT (2021) Sentiment analysis of COVID-19 vaccination from survey responses in Bangladesh
13. Ezen-Can AA (2020) Comparison of LSTM and BERT for small corpus
14. Sood S (2020) LSTM & BERT models for natural language processing (NLP). OpenGenus IQ: Learn Computer Science; Available from: https://iq.opengenus.org/lstm-bert-nlp-model/
15. Naseem U, Razzak I, Khushi M, Eklund PW, Kim J (2021) Covidsenti: a large-scale benchmark Twitter data set for COVID-19 sentiment analysis. IEEE Trans Comput Social Syst 8:1003
16. Roy S, Chakraborty U (2013) Soft computing. Pearson Education India. https://cds.cern.ch/record/1952120
17. Qawqzeh YK, Otoom MM, Al-Fayez F, Almarashdeh I, Alsmadi M, Jaradat G (2019) A proposed decision tree classifier for atherosclerosis prediction and classification. IJCSNS 19(12):197
18. Breiman L (2001) Random forests. Mach Learn 45(1):5–32
19. Singh N, Sharma N, Sharma AK, Juneja A (2017) Sentiment score analysis and topic modelling for GST implementation in India. Soft computing for problem solving. Soft Comput Probl Solving: SocProS 2(817):243
20. Sanh V, Debut L, Chaumond J, Wolf T (2019) DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In: arXiv preprint arXiv:1910.01108.
21. Jacob D, Ming-Wei C, Kenton L, Kristina T (2018) BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805
22. Garbowicz K (2021) DilBERTˆ2: humor detection and sentiment analysis of comic texts using fine-tuned BERT models
23. Yinhan L, Myle O, Naman G, Jingfei D, Mandar J, Danqi C, Omer L, Mike L, Luke Z, Veselin S (2019) RoBERTa: a robustly optimized BERT pretraining approach

24. Li L, Zhang Q, Wang X, Zhang J, Wang T, Gao T-L, Duan W, Tsoi KK-F, Wang F-Y (2020) Characterizing the propagation of situational information in social media during covid-19 epidemic: a case study on weibo. IEEE Trans Comput Social Syst 7:556–562
25. Baker QB, Shatnawi F, Rawashdeh S, Al-Smadi M, Jararweh YJ (2020) Detecting epidemic diseases using sentiment analysis of Arabic tweets. J Univ Comput Sci 26:50–70
26. Jianqiang Z, Xiaolin G, Xuejun Z (2018) Deep convolution neural networks for twitter sentiment analysis. IEEE Access 6:23253–23260
27. Rustam F, Khalid M, Aslam W, Rupapara V, Mehmood A, Choi GS (2021) A performance comparison of supervised machine learning models for Covid-19 tweets sentiment analysis. PLoS ONE 16(2):e0245909
28. Chintalapudi N, Battineni G, Amenta F (2021) Sentimental analysis of COVID-19 tweets using deep learning models. Infect Dis Rep 13(2):329–339
29. Raamkumar AS, Tan SG, Wee HL (2020) Use of health belief model–based deep learning classifiers for covid-19 social media content to examine public perceptions of physical distancing: Model development and case study. JMIR Public Health Surveill 6(3):e20493
30. Sunagar P, Kanavalli A, Poornima V, Hemanth VM, Sreeram K, Shivakumar KS (2021) Classification of Covid-19 tweets using deep learning techniques. Inventive systems and control. Springer, Singapore, pp 123–136
31. Qasim R, Bangyal WH, Alqarni MA, Ali Almazroi A (2022) A fine-tuned BERT-based transfer learning approach for text classification. J Healthcare Eng
32. Jaiman A, Bhandari P Analysis of COVID-Twitter-BERT
33. Basiri ME, Nemati S, Abdar M, Asadi S, Acharrya UR (2021) A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets. Knowl Based Syst 228:107242
34. Samuel J, Ali GG, Rahman M, Esawi E, Samuel Y (2020) COVID-19 public sentiment insights and machine learning for tweets classification. Information 11(6):314
35. Priya A, Garg S, Tigga NP (2020) Predicting anxiety, depression and stress in modern life using machine learning algorithms. Proc Comput Sci 167:1258–1267
36. Kaur H, Ahsaan SU, Alankar B, Chang V (2021) A proposed sentiment analysis deep learning algorithm for analyzing COVID-19 tweets. Inf Syst Front 23(6):1417–1429
37. Catelli R, Gargiulo F, Casola V, De Pietro G, Fujita H, Esposito M (2021) A novel covid-19 data set and an effective deep learning approach for the de-identification of Italian medical records. IEEE Access 9:19097–19110
38. Chandrasekaran G, Hemanth J (2022) Deep learning and TextBlob based sentiment analysis for coronavirus (COVID-19) using twitter data. Int J Artif Intell Tools 31(1):2250011
39. Singh C, Imam T, Wibowo S, Grandhi S (2022) A deep learning approach for sentiment analysis of COVID-19 reviews. Appl Sci 12(8):3709
40. Sunitha D, Patra RK, Babu NV, Suresh A, Gupta SC (2022) Twitter sentiment analysis using ensemble based deep learning model towards COVID-19 in India and European countries. Pattern Recogn Lett 158:164–170
41. Wei J, Zou K (2019) Eda: easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196
42. William YW, Diyi Y (2015) That's so annoying!!!: a lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp 2557–2562
43. Wu B, Liu L, Yang Y, Zheng K, Wang X (2020) Using improved conditional generative adversarial networks to detect social bots on Twitter. IEEE Access 8:36664–36680
44. Wu L, Xia Y, Tian F, Zhao L, Qin T, Lai J, Liu TY (2018) Adversarial neural machine translation. In: Asian Conference on Machine Learning pp 534–549
45. Yang Z, Chen W, Wang F, Xu B (2017) Improving neural machine translation with conditional sequence generative adversarial nets. arXiv preprint arXiv:1703.04887
46. Munková D, Munk M, Vozár M (2013) Data pre-processing evaluation for text mining: transaction/sequence model. Proc Comput Sci 18:1198–1207
47. Vijayarani S, Ilamathi MJ, Nithya M (2015) Preprocessing techniques for text mining-an overview. Int J Comput Sci Commun Netw 5(1):7–16
48. Le QV, Mikolov T (2014) Distributed representations of sentences and documents. Int Conf Mach Learn 4(2):1188
49. Haque MM, Biswas N, Roy NS, Rafi AH, Islam SU, Lubaba SS, Rahman RM (2021) Data mining techniques to categorize single paragraph-formed self-narrated stories. ICT analysis and applications. Springer, Singapore, pp 701–713
50. Nils R, Iryna G (2019) SentenceBERT: sentence embeddings using siamese BERTNetworks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th

International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, pp 3982–3992
51. Florian S, Dmitry K, James P (2015) FaceNet: a unified embedding for face recognition and clustering. arXiv preprint arXiv:1503.03832
52. Daniel C, Yinfei Y, Sheng-yi K, Nan H, Nicole L, Rhomni J, Noah C, Mario G-C, Steve Y, Chris T, Yun-Hsuan S, Brian S, Ray K (2018) Universal sentence encoder. arXiv preprint arXiv:1803.11175
53. Iyyer M, Manjunatha V, Boyd-Graber J, Daumé III H (2015) Deep unordered composition rivals syntactic methods for text classification. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing vol 1, pp 1681–1691
54. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems pp 5998–6008
55. The Illustrated Transformer – Jay Alammar – Visualizing machine learning one concept at a time, Github. [Online]. http://jalammar.github.io/illustrated-transformer/%0A. (Accessed: 20 Dec 2021)
56. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
57. Kadhim AI (2019) Survey on supervised machine learning techniques for automatic text classification. Artif Intell Rev 52(1):273–292. https://doi.org/10.1007/s10462-018-09677-1
58. Kour H, Gupta MK (2022) An hybrid deep learning approach for depression prediction from user tweets using feature-rich CNN and bi-directional LSTM. Multimed Tools Appl 81:23649–23685
59. Vinyals O, Toshev A, Bengio S, Erhan D (2015) Show and tell: a neural image caption generator. In: CVPR
60. Seo J, Yoo K, Choi S et al (2019) The latent learning model to derive semantic relations of words from unstructured text data in social media. Multim Tools Appl 78:28649–28663
61. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
62. “Amazon Reviews for Sentiment Analysis.” [Online]. https://www.kaggle.com/bittlingmayer/amazonreviews.
63. “Suicide and depression detection using subreddit and reddit platform” [online]. https://www.kaggle.com/nikhileswarkomati/suicide-watch.
64. “Sentinet140 dataset for Sentiment Analysis” [online]. https://www.kaggle.com/kazanova/sentiment140.
65. Wang C, Jiang F, Yang H (2017) A hybrid framework for text modeling with convolutional RNN. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining pp 2061–2069

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.