Hybrid Deep Learning Model for COVID-19 Prediction Using Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (LSTM) Network

Deepti Malhotra* and Gurinder Kaur Sodhi

***Abstract*—In a very short interval, COVID-19 had spread all over the world and affected the medical and economic condition of all the developed, developing and underdeveloped countries, badly. Though it has been more than three years since its first onset, but even today its fear grips the entire world. Time and again, its various mutants have been detected. Early-stage diagnosis, reporting and isolation of the patient are the only ways to restrict the spread of virus of this nature. In addition, prediction models to help in this regard. This paper presents a novel Hybrid Model based on Deep Learning techniques. In this work, an authentic dataset has been collected, pre-processed and finally classified for the prediction of the disease. The model has been implemented using Python programming code. It's accuracy, precision and recall has been calculated, and compared with that of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) model.**

***Index Terms*—COVID-19, deep learning, Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), bidirectional LSTM, hybrid model**

I. INTRODUCTION

In December 2019, the first case of infectious coronavirus disease (COVID-19) was reported in the Wuhan province of China. Within no time it engulfed the entire world. While some of the infected persons get cured easily without any specific treatment, for others, the situation may vary. The virus mainly targets the person's respiratory system. However, in acute cases it effects the entire body's organs and system. Elderly, pregnant females, cancer patients, individuals with existent comorbidities like: heart patients, diabetic people, patients with respiratory disease, are at a higher risk for catching this virus [1]. Driven by its prevailing inflammatory response, studies show that the virus affects nearly all organs of the body. Approximate 10–15% of COVID-19 patients may experience COVID-19 for a long time effecting the heart, lungs and nervous system. Transmission of corona virus mainly occurs through droplets in the air from an infected person's speaking, coughing or sneezing or even by touching effected objects or parts. According to the World Health Organization (WHO), repetitive hand-washing, disinfecting, social distancing, mask-wearing and avoiding touching one's face are the major measures to prevent oneself from getting infected. The list of most common symptoms of COVID-19 released by WHO includes fever, dry cough and fatigue, while trivial symptoms include headache, sore throat, diarrhoea, conjunctivitis, loss of smell or rashes. Some of the severe signs include breathlessness, pain in chest and loss of speech and mobility [2].

Despite the government, community and individual's efforts, the virus has spread globally, sparing no place. This has undoubtedly put a huge strain on the major resources i.e., hospitals, necessary medical equipment, healthcare workers and testing kits too. However, early diagnosis of the disease and giving the possible care to suspected patient is something that can be done easily. Hence, a COVID-19 automated prediction system is urgently required in the present scenario to detect the virus presence. Machine Learning (ML) classification algorithms, datasets and software are fundamental requirements for developing prediction framework. Fig. 1 represents a generic COVID-19 prediction model [3].

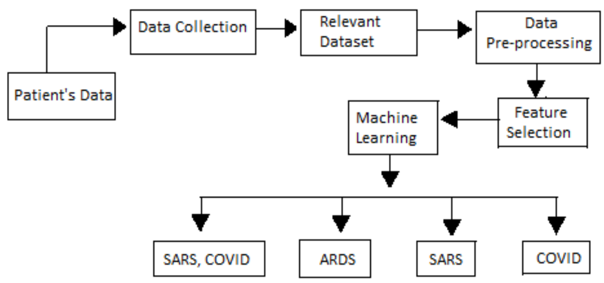

Fig. 1. Generic COVID-19 prediction architecture.

The stages in this predictive framework are: Data collection, Pre-processing, Feature Selection and Machine learning [4–7].

II. COVID-19 OUTBREAK PREDICTION MODELS

Following is the list of some commonly used COVID-19 outbreak prediction models.

1. Decision tree: A decision tree is an algorithm that creates a graphical tree-like configuration. To classify cases this algorithm uses a root node containing a test condition (e.g., whether the person has a sore throat) and branches that provide answers, labels or the supposed class [8, 9].

2. Random Forest (RF): Similar to decision tree, the RF algorithm also builds a tree, but in large numbers. The formation of trees depends on the values of random samples in the dataset and the final result relies on the results of the majority of the established trees [10].

3. Naive Bayes: It is a statistical supervised ML algorithm employs for predicting the possibilities of class membership. The Naive Bayes (NB) offers a higher accuracy on an enormous dataset as well as provides optimal performance on

Manuscript received November 30, 2022; revised January 11, 2023; accepted March 27, 2023.

Deepti Malhotra and Gurinder Kaur Sodhi are with Desh Bhagat University, Punjab, India.

*Correspondence: deeptimalhotra1981@gmail.com (D.M.)

DOI: 10.7763/IJCTE.2023.V15.1341

small datasets [11, 12].

4. K-Nearest Neighbours (KNN): It is a simple and standard supervised ML algorithm. KNN algorithm is useful for classifying a given instance on the basis of the majority of the classes among its nearest neighbour which are available in the dataset [13, 14].

5. Long Short-Term Memory (LSTM): The LSTM model is an improved version of Recurrent Neural Networks (RNN) and its range extends to multiple domains such as text recognition, finance and industrial engineering. LSTM has an input layer, an output layer and hidden layers [15–17].

6. Convolutional Neural Networks (CNN): A type of Deep Neural Network, it is based on the visual system of the human brain. It is based on the mathematical operation of convolution and primarily constitutes of three layers: convolution, pooling and a fully connected layer [18].

III. Research Methodology

The focus of this research work is to predict COVID-19 disease. The prediction task consists of various phases namely dataset collection, pre-processing, and classification. A hybrid deep learning model is designed in this work which is the combination of CNN and Bidirectional LSTM. LSTM networks make use of a micro-gate control by combining short-term and long-term and provide solution of gradients disappearing to some level. It consists of three explicit frameworks: a forget gate, an input gate and an output gate. LSTM, by means of a structure termed as gate may remove or add information to the cell state. In the first step, the information that is to be discarded from the cell state is determined. This purpose is solved through a layer called the forget gate. The forget gate after reading the hidden state of the earlier moment h_{t-1} and the current input data x_t, generates output as a vector between 0 and 1. The value between 0 and 1 in this vector pint outs the scale of information reserved or rejected in the cell state c_t. The value 0 denotes that all information is rejected while 1 indicates the retention of all information.

$$z_f = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{1}$$

The second step aims to determine the scale of fresh information added to the cell state. This is a two-step process. This step initially decides the information to be updated through an input gate operation by using h_{t-1} and x_t. Then, h_{t-1} and x_t are used for obtaining a new candidate cell state z via a tanh layer.

$$z_i = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

$$z = tanh(W \cdot [h_{t-1}, x_t] + b) \tag{3}$$

Next, the cell state is updated as follow:

$$c_t = z_f \odot c_{t-1} + z_i \odot z. \tag{4}$$

The final step is to determine the output value. Once the cell state is updated, the part of the cell state which will be the output depending upon the input h_{t-1} and x_t, is decided [19]. For this purpose, the input must pass through a sigmoid layer known as output gate so that the judgment conditions can be obtained. After this, there is the need to pass the cell state via the tanh layer to get a vector between −1 and 1. Finally, the output is achieved by multiplying this vector by the analysis conditions attained by the output gate.

$$z_o = \sigma(W_O \cdot [h_{t-1}, x_t] + b_0) \tag{5}$$

$$h_t = z_o \odot tanh(c_t) \tag{6}$$

In the above expression, z_i is the forget gate, z_f is the input gate, and z_o is the output gate. Apart from this, z denotes the input via a tanh layer. This is also termed as candidate cell state. At last, $\odot$ represents a multiplication function of the related components of the matrix. Fig. 2 is a screenshot of the various features used while working in python using this technique.

CNN is a Deep Learning algorithm in the area of Prediction. Similar to other ML algorithms, this algorithm is capable of learning from the metrics to which a training set is employed so that least empirical and structural risk is obtained such as Loss Function (LF) is alleviated. Diverse operations support distinct LFs that implies the error occurred in predictive values and value having label of correct. DL is useful to stack the layers of metrics and establishes an association amid them and activation function. After that, the network becomes adaptable for the nonlinear functions which have complexity. Unlike the traditional neural networks, there are three layers for building the Convolutional Neural Network which are convolutional, pooling and fully connected layers [20, 21].

conv1d_26 (Conv1D)	(None, 24, 64)	384
activation_30 (Activation)	(None, 24, 64)	0
dropout_13 (Dropout)	(None, 24, 64)	0
max_pooling1d_10 (MaxPooling1D)	(None, 6, 64)	0
conv1d_27 (Conv1D)	(None, 6, 128)	41088
activation_31 (Activation)	(None, 6, 128)	0
dropout_14 (Dropout)	(None, 6, 128)	0
max_pooling1d_11 (MaxPooling1D)	(None, 1, 128)	0
conv1d_28 (Conv1D)	(None, 1, 256)	164096
activation_32 (Activation)	(None, 1, 256)	0
dropout_15 (Dropout)	(None, 1, 256)	0
bidirectional (Bidirectional)	(None, 128)	164352
dense_8 (Dense)	(None, 2)	258
activation_33 (Activation)	(None, 2)	0

Total params: 370,178

Fig. 2. Model description.

The initial layer is employed for mapping an input of multilayer into an output. Every image layer is considered as a channel. The mapping task is accomplished using a kernel for every channel [18]. A convolution is exploited with the kernel with the *purpose* of separating every layer into small units (5 × 5) and generating a number from every unit. The convolution of every unit is expressed as:

$$E_{p,q} = \sum_{i,j} A_{ij} \cdot W_{ij} \quad (7)$$

In this, a square matrix based on (p, q) is represented with A, the correspondence components are defined through A_{ij} and W_{ij} and the output is obtained in the form of $E_{p,q}$. The sigmoid function or rectified linear unit function plays a role of an activation function to make the mapping non-linear. The given equation defines the rectified linear unit:

$$f(x) = \begin{cases} x, & x >= 0 \\ 0, & x < 0 \end{cases}$$

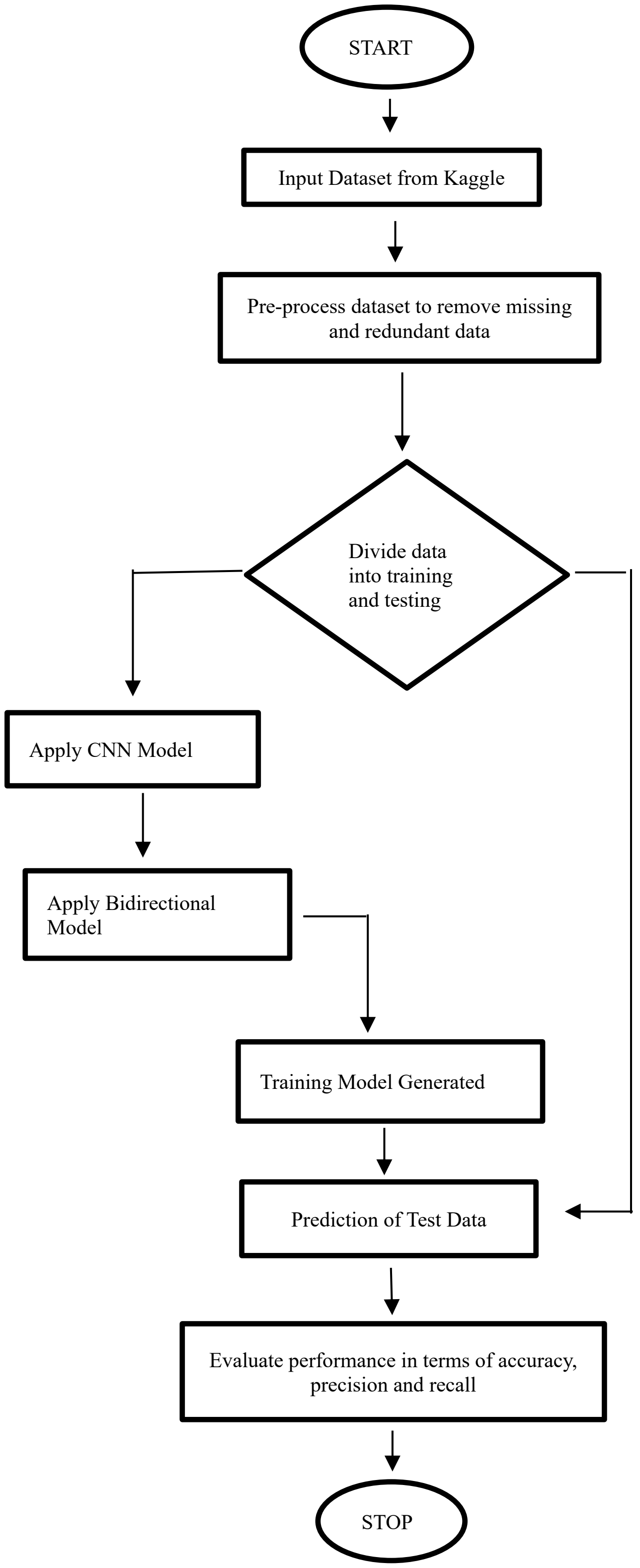

Fig. 3. Proposed flowchart.

This layer aims to abstract the classic attributes from the input. The lower layers are executed for retrieving the horizontal or vertical edges and the upper layers emphasize on integrating these attributes.

Fig. 3 shows the methodology followed in the present work for COVID-19 prediction using Deep Learning based hybrid model.

IV. Result and Discussion

The main objective of this research is to predict COVID-19 disease by applying deep learning model. The proposed hybrid deep learning model will be compared with deep learning models like CNN, LSTM. The dataset is collected from Kaggle which is from Mexico. The dataset includes information on positive and negative cases reported by the General Directorate of Epidemiology, the Secretariat of Health in Mexico. The reported results of RT-PCR testing dataset have been used to generate this dataset. The dataset contains the target set which defines the positive and negative cases and will vary between 0 and 1. In COVID-19 prediction, 0 value is for negative case and 1 is for positive case.

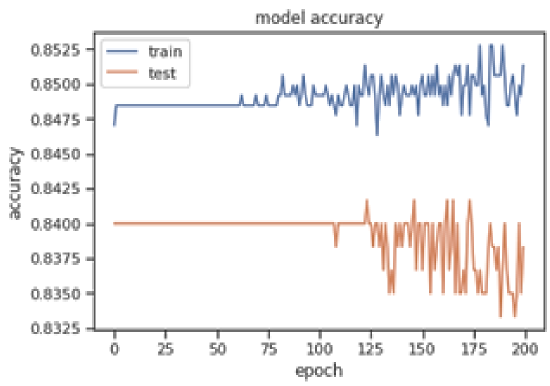

Fig. 4. Model accuracy.

As shown in Fig. 4, the training accuracy of the hybrid deep learning model is approx. 85% and on the other side test accuracy of the model is approx. 84% in prediction.

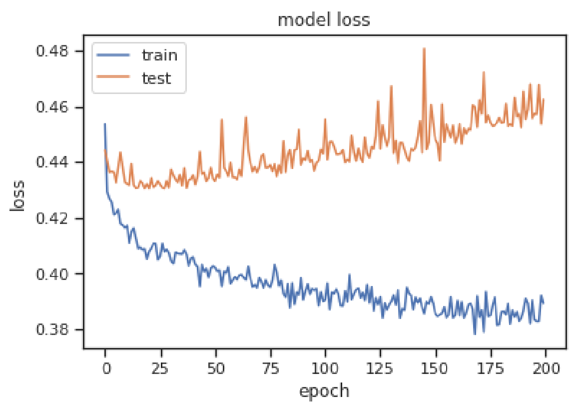

Fig. 5. Model loss.

As shown in Fig. 5, the model loss of the training data is approx. 34% and testing model loss is high which is 46%.

The performance of a classification model is assessed in terms of accuracy, precision and recall. These are defined as:

1. Accuracy: It is the ratio of correctly predicted observations to all the observations.

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}}$$

2. Precision: This is the ratio of accurate positive predictions to the total number of positive predictions.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

3. Recall: It is the ratio of accurate positive predictions to all the predicted results.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

In this section, the above-mentioned parameters have been calculated for CNN, LSTM and Proposed Model. The results of the same are presented below:

TABLE I: CNN MODEL RESULTS

Class	Accuracy	Precision	Recall
1 (Affected)	0.78	0.77	0.78
0 (Non-affected)	0.8	0.03	0.06

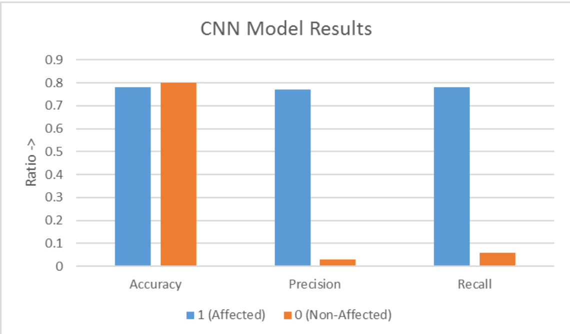

Fig. 6. CNN model result analysis.

TABLE II: LSTM MODEL RESULTS

Class	Accuracy	Precision	Recall
1 (Affected)	0.80	0.81	0.80
0 (Non-affected)	0.09	0.03	0.07

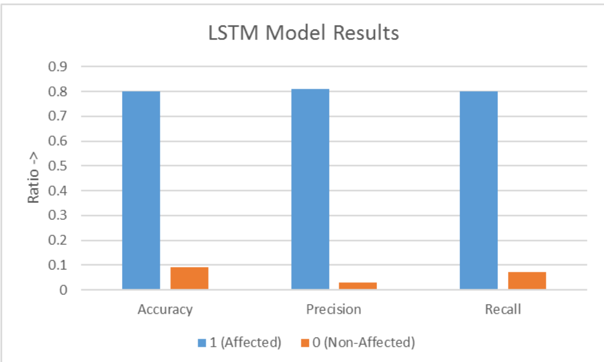

Fig. 7. LSMT model result analysis.

TABLE III: HYBRID DEEP LEARNING RESULTS

Class	Accuracy	Precision	Recall
1 (Affected)	0.84	0.84	0.83
0 (Non-affected)	0.16	0.06	0.07

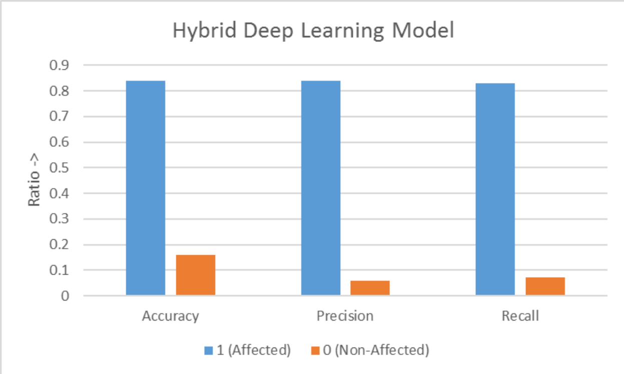

Fig. 8. Hybrid deep learning model.

The model has been trained and tested subsequently using Python. The results have been presented in Tables I–IV. Table I represents the value of accuracy, precision and recall for the CNN model for affected and non-affected cases. The same has been presented in Table II for LSTM model and for proposed hybrid model in Table III. Figs. 6–8 are the graphical representation of the python's calculated results of accuracy, precision and recall for the CNN, LSTM and proposed hybrid model respectively. Table IV and Fig. 9 clearly depict better performance in terms of these results.

TABLE IV: RESULT COMPARISON

Model	Accuracy	Precision	Recall
CNN	0.78	0.77	0.78
LSMT	0.80	0.81	0.80
Hybrid	0.84	0.84	0.83

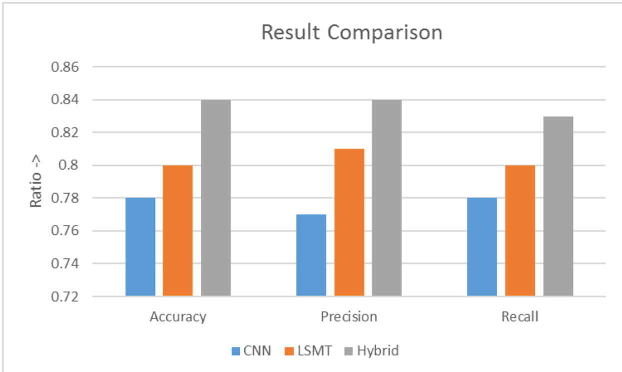

Fig 9. Result comparison.

V. CONCLUSION

There have been more than million causalities due to COVID-19, with thousands of infection cases and deaths daily. Preventing the spread of COVID-19 infection has become a main challenge for researchers around the world. The motivation of the proposed method is clearly to aim for an accurate prediction of COVID-19. In the case of COVID-19, we cannot take a risk of 'False Negative' in spite of the patient being infected [22–24]. We all understand its dire repercussions. Hence, we propose a novel Deep Learning Model i.e., CNN and Bidirectional LSTM. The accuracy of both have been expressed in terms of recall, and precision, and comparison to a Hybrid Model. The proposed model achieves an accuracy of 84%, which is 3% higher than the existing CNN, LSTM Models. In the future, transformers can be used in the proposed model to increase accuracy.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

The research work and paper writing has been done by Deepti Malhotra under the guidance of Gurinder Kaur Sodhi. The final version has been approved by both the authors.

References

[1] Z. A. A. Alwaeli and A. A. Ibrahim, "Predicting COVID-19 trajectory using machine learning," in *Proc. 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 2020.

[2] S. Bodapati, H. Bandarupally, and M. Trupthi, "COVID-19 time series forecasting of daily cases, deaths caused and recovered cases using long short term memory networks," in *Proc. IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, 2020.

[3] H. Chao, X. Fang, and P. Yan, "Integrative analysis for COVID-19 patient outcome prediction," *Medical Image Analysis*, 2020.

[4] D. Rafiq, S. A. Suhail, and M. A. Bazaz, "Evaluation and prediction of COVID-19 in India: A case study of worst hit states," *Chaos, Solitons & Fractals*, 2020.

[5] S. F. Ardabili, A. Mosavi, P. Ghamisi, F. Ferdinand, A. R. Varkonyi-Koczy, U. Reuter, T. Rabczuk, and P. M. Atkinson, "COVID-19 outbreak prediction with machine learning," *Algorithms*, vol. 13, no. 249, 2020.

[6] F. Petropoulos and S. Makridakis, "Forecasting the novel coronavirus COVID-19," *PLoS ONE*, vol. 15, no. 3, e0231236, 2020.

[7] Z. Yang, Z. Zeng, K. Wang, S.-S. Wong, W. Liang, M. Zanin *et al.*, "Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions," *Journal of Thoracic Disease*, 2020.

[8] P. Nadella, A. Swaminathan, and S. V. Subramanian, "Forecasting efforts from prior epidemics and COVID 19 predictions," *European Journal of Epidemiology*, vol. 35, pp. 727–729, 2020.

[9] P. Kumari and D. Toshniwal, "Real-time estimation of COVID-19 cases using machine learning and mathematical models—The case of India," in *Proc. IEEE 15th International Conference on Industrial and Information Systems (ICIIS)*, 2020.

[10] G. Giordano, F. Blanchini, R. Bruno, P. Colaneri, A. D. Filippo, A. D. Matteo, and M. Colaneri, "Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy," *Nature Medicine*, vol. 26, pp. 855–860, 2020.

[11] N. P. Jewell, J. A. Lewnard, and B. L. Jewell, "Predictive mathematical models of the COVID-19 pandemic underlying principles and value of projections," *JAMA*, vol. 323, no. 19, 2020.

[12] W. Zou, "The COVID-19 pandemic prediction in the US based on machine learning," in *Proc. International Conference on Public Health and Data Science (ICPHDS)*, 2020.

[13] S. Shoer, T. Karady, A. Keshet, V. Shalev, R. Balicer, and E. Sega, "A prediction model to prioritize individuals for a SARS-CoV-2 test built from national symptom surveys," *Clinical and Translational Resource and Technology Insights*, 2020.

[14] N. M. Ferguson, D. Laydon, G. Nedjati-Gilani, N. Imai *et.al*., "Impact of Non-Pharmaceutical Interventions (NPIs) to reduce COVID-19 mortality and healthcare demand," *Imperial College COVID-19 Response Team*, 2020.

[15] R. Gupta, G. Pandey, P. Chaudhary, and S. K. Pal, "Machine learning models for government to predict COVID-19 outbreak," *Digital Government: Research and Practice*, vol. 1, no. 4, p. 26, 2020.

[16] S. F. Ardabili, A. Mosavi, P. Ghamisi, F. Ferdinand, A. R. Varkonyi-Koczy, U. Reuter, T. Rabczuk, and P. M. Atkinson, "COVID-19 outbreak prediction with machine learning," *Algorithms*, vol. 13, no. 10, 249, 2020.

[17] D. W. Challener, S. C. Dowdy, and J. C. O'Horo, "Analytics and prediction modeling during the COVID-19 pandemic," *Perspective and Controversy*, vol. 95, no. 9, pp. 8–10, 2020.

[18] A. A. Reshi, F. Rustam, A. Mehmood, A. Alhossan, Z. Alrabiah, A. Ahmad, H. Alsuwailem, and G. S. Choi, "An efficient CNN model for COVID-19 disease detection based on X-Ray image classification," *Complexity Journal by WILEY Hindawi, Special Issue on Complexity and Robustness Trade-Off for Traditional and Deep Models*, vol. 2021, 6621607, 2021.

[19] E. Sevinç, "An empowered AdaBoost algorithm implementation: A COVID-19 dataset study," *Computers & Industrial Engineering*, vol. 165, 107912, 2022.

[20] M. Adhikari, M. Ambigavathi, V. G. Menon, and M. Hammoudeh, "Random forest for data aggregation to monitor and predict COVID-19 using edge networks," *IEEE Internet of Things Magazine*, vol. 4 issue 2, pp. 40–44, 2021.

[21] E. Gothai, R. Thamilselvan, R. R. Rajalaxmi, R. M. Sadana, A. Ragavi, and R. Sakthivel, "Prediction of COVID-19 growth and trend using machine learning approach," *Mater Today Proc.*, 2021, doi: 10.1016/j.matpr.2021.04.051

[22] M. Sahinoglu and H. Sahinoglu, "Consequences and lessons from 2020 pandemic disaster: Game-theoretic recalibration of COVID-19 to mobilize and vaccinate by rectifying false negatives and false positives," *International Journal of Computer Theory and Engineering*, vol. 14, no. 3, pp. 109–125, 2022.

[23] G. Pena, J. Gambini, and N. R. Barraza, "Identifying the most relevant attributes to explain peaks of COVID-19 infections and deaths by machine learning methods," *International Journal of Computer Theory and Engineering*, vol. 15, no. 1, pp. 1–9, 2023.

[24] M. Sahinoglu and A. Arteta, "Editorial for the special issue (2022) on 'The recent advances in computer theory and software engineering on COVID-19 pandemic and incurable killer diseases'," *International Journal of Computer Theory and Engineering*, vol. 15, no. 1, p. 61, 2023.