

Chapter 3

Impact of Internet of Health Things (IoHT) on COVID-19 Disease Detection and Its Treatment Using Single Hidden Layer Feed Forward Neural Networks (SIFN)



S. Murugan, K. Vijayakumar, V. Sivakumar, R. Manikandan, Ambeshwar Kumar, and K. Saikumar

Abstract COVID-19 endemic has made the entire world face an extraordinary challenging situation which has made life in this world a fearsome halt and demanding numerous lives. As it has spread across 212 nations and territories and the infected cases and deaths are increased to 5,212,172 and 334,915 (as of May 22 2020). Still, it is a real hazard to human health. Severe Acute Respiratory Syndrome cause vast negative impacts economy and health populations. Professionals involved in COVID test can commit mistakes when testing for identifying the disease. Evaluating and diagnosing the disease by medical experts are the significant key factor. Technologies like machine learning and data mining helps substantially to increase the accuracy of identifying COVID. Artificial Neural Networks (ANN) has been extensively used for diagnosis. Proposed Single Hidden Layer Feedforward Neural Networks (SLFN)-COVID approach is used to detect COVID-19 for disease

S. Murugan (✉)

Department of Computer Science, Sri Aravindar Engineering College,
Villupuram, Tamil Nadu, India
e-mail: smartrugans@gmail.com

K. Vijayakumar

Department of Software Engineering, SRM Institute of Science and Technology,
Chennai, Tamil Nadu, India

V. Sivakumar

School of Computing, Department of Computer Science and Engineering, Vel Tech
Rangarajan Dr. Sagunthala R&D Institute of Science and Technology,
Chennai, Tamil Nadu, India

R. Manikandan · A. Kumar

School of Computing, SASTRA Deemed University, Thanjavur, Tamil Nadu, India

K. Saikumar

Research Scholar (JRF), KLEF, Guntur, Andhra Pradesh, India

detection on creating the social impacts and its used for treatment. The experimental results of the proposed method outperforms well when compared to existing methods which achieves 83% of accuracy, 73% of precision, 68% of Recall, 82% of F1-Score.

Keywords COVID-19 · Machine learning · Single Hidden Layer Feed forward Neural Networks

3.1 Disease Detection Using IoHT

The IoT has changed the way we live, work and take choices. In several industries, including health care, it has already had a substantial and positive influence. A huge revolution has begun in the health sector with the impact of IoT. A new layer in health care infrastructure, Internet of Health Things (IoHT), has been created by rapid developments in sensor-based applications and Internet technologies [1]. IoHT is a term that represents internet-connected identifiable devices and which can be employed in the medical sector. Through medical attention and detection errors, these technologies promote safety. The data collected by gadgets are analysed and medical consultations are remotely conducted. The Web of Things has changed the world already. The life and work of people has been influenced.

In big data, essential source is healthcare. There is a strong need for early identification of disease for accurate review of healthcare data. Viruses are small organisms that only reproduce within an organism's living cells. In living species, coronaviruses are one of most common families of viruses that cause different respiratory diseases. In Wuhan, China, an increasing number of rare cases of pneumonia were recorded in December of 2019. A novel coronavirus, known as COVID-19, was quickly determined to be cause of this epidemic.

Coronaviruses, a kind of viruses, causes severe sickness and disorders in human health. This sickness was burst only at the end of December 2019. This new virus which has spread worldwide is very infectious. WHO (World Health Organization), on 30th January 2020, declared this as a PHEIC (Public Health Emergency of International Concern) because it spread around eighteen nations. On 11th of February 2020, WHO named this outbreak as "COVID-19". Apart from China, , as there was a tremendous increase in COVID-19 cases which was about thirteen times than earlier, which was more than 118,000 in 114 countries and caused 4000 deaths, on March 11th 2020, WHO announced this a pandemic [2].

COVID-19 along with ARDS, SARS (Severe Acute Respiratory Syndrome) is a family of viruses. WHO mentions that this virus which is transmitted through the respiratory tract affects an individual who is in contact with the infected individual [3]. The symptoms are observed from the 2nd day till 14th day, which is the development duration of MERS, and SARS. Specialist face fundamental difficulties while sharing the information during disorder recognizing and to give treatment [4]. Individual with COVID symptoms are keenly observed. Specifically, no treatment

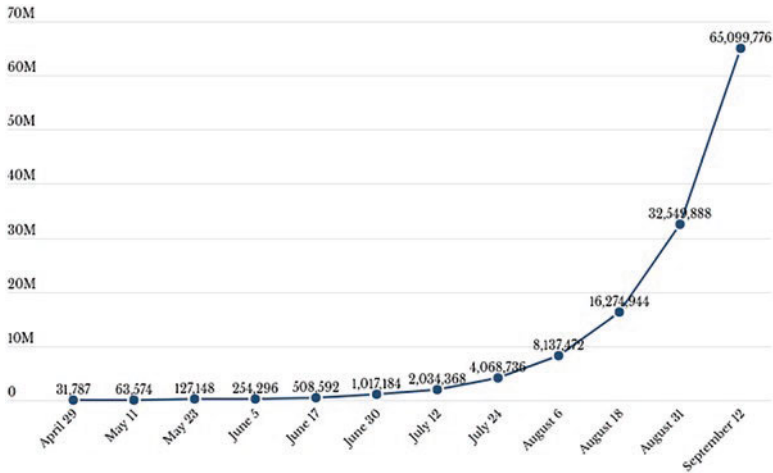


Fig. 3.1 Growth of COVID-19 in India

has been yet found as on April 10th 2020 to prevent this dreadful virus, and thus patients are given treatment based on their symptoms. Medicines, namely hydroxy-chloriquine, anti-virals, antipyretic are used for treatment. But some preventive measures are being followed like frequently washing hands with soap for a minimum of 20 seconds, avoiding very closer contact among individuals thereby maintaining a distance of about 1 meter, wearing mask which covers nose and mouth which is believed that there are chances to reduce the infection being spread [5]. In brief, reliable and fast early detection of confirmed COVID-19 cases plays vital role in prompt quarantine and medical care, which is also of great significance for patient prognosis, disease prevention, and public health protection. If this continuous, India could possibly face more than 65 million cases by the end of September 2020. COVID-19 has rapidly grown in India as illustrated in Fig. 3.1.

Apart from the procedures followed by clinics, machine learning offers more support to recognize the disease by processing textual and image information of individuals but requires enormous data. Moreover, the nature of the virus can also be predicted and classified. Supervised machine learning approaches requires annotated data to classify text or image data. Algorithms based on this approach are used in various critical projects. Outbreak of pandemic of COVID-19 has brought many countries to a halt. Based on number of affected nations, cases and applied digital measures around world, COVID-19 pandemic can be distinguished. Restricted information is also accessible about how digital interventions can help to contain a global pandemic. Current pandemic situation has kindled several researchers worldwide to overcome this problem. Internet of Things (IoT) is a networked device/operation system that complies with all network aspects, for example hardware, software and network connectivity, as well as all other essential electronic/computer systems. If we talk a little more about IoT, it goes beyond a notion which builds the

entire architectural context that allows data to be integrated and exchanged effectively between the data subjects and providers. Most of the problems arise in current typical situation as the non-effective reachability of patients, which after concern about vaccine development is second most significant difficulty. Utilisation of IoT concept makes it quite valuable for patients to be accessible, which ultimately helps them to receive considerable care to get out of sickness. This work is arranged as: Sect. 3.2 describes the background study, Sect. 3.3 discusses methodology, Sect. 3.4 depicts performance analysis and concludes with Sect. 3.5.

3.2 State of Art Methods

Iwendi Celestine et al. 2020 introduced improved Random Forest approach enhanced by the AdaBoost approach. This model used travel, geographical, demographic, and health information of the persons infected by COVID-19 virus for predicting the severity and recovery [6].

Tuli et al. 2020 designed a better mathematical model for predicting and analyzing development and threat of COVID-19 in nations globally. When iterative weighting to fit Generalized Inverse Weibull distribution was involved, a best fit was achieved for developing a prediction model which was deployed on a cloud platform in order to obtain more precise and real-time prediction results [7].

Sujath et al. 2020 developed useful method for forecasting spread of COVID virus. For additional processing in the future, details of the patients should be persistently maintained [8]. Jiang et al. 2020 implemented a machine learning approach which predicted an individual who was infected by COVID virus and there was a possibility for developing ARDS. An accuracy of 80% was obtained [9].

Abdani et al. 2020 suggested and implemented a lightweight deep learning approach to accurately detect the risk of COVID-19 [10]. This model was designed such that it can be deployed on different platforms like tablets, mobiles, and normal computers with no distress of memory capacity. This approach was developed by integrating modified spatial pyramid pooling (SPP) model [11] with convolutional neural network (CNN). Through the use of modified SPP called SPP-COVID-Net, numerous inputs were processed simultaneously [12] with a total parameters of 862,331 with not more than 4 MB of storage. The multi-scale facility of this model paved way for identifying COVID patients with different severity levels. This SPP-COVID-Net achieved the average accuracy of about 0.946. In [13], an online healthcare surveillance model is proposed that is capable of evaluating patient health data in order to negotiate occurrence of death. This collects associated patient information required by sensing and healthcare instruments for investigation.

Olivera et al. 2019 [14] model based on the probabilistic principle of Bayesian Monte Carlo to forecast distribution of infectious diseases is suggested. Big downside to this strategy is that it is impacted by collection of past as well as subsequent probabilities. Aayush Jaiswal et al. [15] use of pre-trained deep learning

architectures to classify and diagnose COVID-19 in chest CT as an automated tool. By utilizing its own trained weights on ImageNet dataset along with a convolutionary neural structure, proposed model is used to extract functions.

Tushaar Gangavarapu et al. [16] To achieve most discriminative and instructive function subspace, developed novel filter-wrapper hybrid ensemble feature selection technique depends on weighted incidence frequency and penalty method. Longxi Zhou et al. [17] propose fully automated, fast, precise, and machine-agnostic system that can segment and calculate CT scans from various sources to the infection regions.

Shreyas Setlur Arun et al. [18] prediction of spread of COVID19 disease is presented and scale of pandemic, recovery rate and fatality rate are expected. Yujin Oh et al. [19] patch-based method with a comparatively small number of trainable COVID-19 diagnostic parameters for CNN is proposed. Our statistical analysis of possible imaging biomarkers of CXR radiographs inspires suggested method.

Vasilis Z. Marmarelis [20] A novel method, each defined by a Riccati equation with adaptively approximate parameters, was proposed based on data-guided identification and concatenation of infection waves. Barabas J et al. [21] presents a device that encapsulates all of above features through readily accessible components (both hardware as well as software) and is again improved with preliminary acquisition of RTLS data, enabling identification of person-to-person communication detected after symptoms to identify potential vectors of infection and mitigate their further spread via smart quarantine.

Deng-Ping Fan et al. [22] to automatically classify contaminated regions from chest CT slices, a novel COVID-19 Inf-Net (Lung Infection Segmentation Deep Network) was proposed. Yanfang Ye et al. [23] propose and develop an AI-driven system called alpha-satellite) to include as an initial providing a dynamic COVID-19 risk assessment in United States. More precisely, device would automatically have risk indices associated with it in a hierarchical way (e.g., state, county, POI) provided a point of interest (POI) to encourage citizens to pick acceptable safeguard actions while mitigating everyday life disturbances. Loveleen Gaur et al. [24] is a realistic method for detection of COVID-19 from chest radiographs and for differentiating from normal and Viral Pneumonia affected by DCNN. In this work, transfer learning evaluates three pretrained models of CNN (EfficientNetB0, VGG16 and InceptionV3). A balance of precision and efficiency with fewer parameters ideal for mobile applications justifies adopting this specific models. Data collection utilised for study is accessible to public and compiled from many sources.

3.3 System Model

The proposed method is used to detect the COVID and diagnoses the disease. The original data is collected from the database, the data is preprocessed by using smoothing method. Once the data is preprocessed then it can be forwarded to feature extraction process. GLCM based feature extraction is used, the classifier input

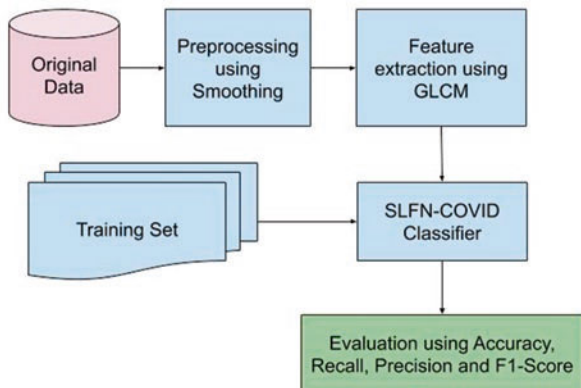


Fig. 3.2 Flow of proposed method

is extracted features that can be SLFN COVID evaluate system. Parameters used to evaluate system is accuracy, recall, F1-Score and precision. Flow of method proposed is depicted in Fig. 3.2.

3.3.1 *Preprocessing Using Smoothing*

It is tedious to process a raw image. Hence, prior to processing, some operations has to be performed like noise removal and ignoring unnecessary portions of the image. This process is termed as preprocessing which further helps to process the image successfully. Moreover, the image quality is improved by smoothing. After this process, noise is reduced, enhanced and the image is reconstructed for extracting salient features to process further. This process is necessary before feature extraction and analysis.

3.3.2 *Grey Level Co-occurrence Matrix (GLCM) Based Feature Extraction*

GLCM is formed with several statistical features obtained from a grey scale image. It is a square matrix given as $G(i, j)$ which involves four directions namely horizontal, vertical, left and right cross directions for partitioning G into the normalized typical matrices. For every adjacent direction, evaluation is performed. These texture features are described as correlation, angular secondary moment, inverse difference moment, contrast, sum of squares, sum average, variance, sum and difference variance, auto correlation, entropy, sum and difference entropy, information measures of correlation 1 and correlation 2, inverse difference, cluster shade and prominence, dissimilarity, and maximum probability.

3.3.3 Classification

The classification using artificial neural network that can be discussed below.

3.3.4 Back Propagation Artificial Neural Network (BPANN)

Currently most common NN model in use is BP neural network [25]. In 1986, Rumelhart and Mclelland proposed BP algorithm that solved weight adjustment issues of nonlinear continuous function in multilayer FFNN area well. It is a standard error algorithm for back propagation [26, 27]. After advent of BP neural network, a lot of research has been performed on activation function selection, structure parameter design, and network defect improvement. Grossberg discovered in 1973 that sigmoid function is very analogous to working conditions of biological neurons, so he began researching relationship between properties of sigmoid function and integrity of NN. He marketed feature to become most commonly deployed BP-NN activation feature.

Core principle of BP method is to break learning process into two steps i.e. constructive signal signalling and error back-propagation. Data is collected from input layer to output layer via hidden layer at point of constructive communication [28]. In output side, output signal forms. In course of sending signal forward network weights are fixed. State of neurons of every layer only influences state of next layer neurons. If it is not feasible to produce desired output in output layer, so error signal can spread again. In back propagation process, step by step, error signal that failed to fulfil accuracy criterion spreads forward and all units of each layer share error. Based on error signal, connection weights are adjusted dynamically. In loop of forward and back change, weight attribute of neurons keeps correcting. Learning ceases until the output signal error meets the precision requirement [29].

3.3.4.1 Topology Structure of BP Neutral Network

As is seen in Fig. 3.3, simplest BPNN has three layers. Input layers, Hidden layers and output layers are used. Number of nodes in input layer is equal to n-input vector dimension. In output layer, number of nodes is proportional to form of output module q. In hidden layer p, number of nodes is correlated with a particular programme that is normally chosen by best.

W_1 , which is defined as a formula (3.1), is relation weight matrix between input layer nodes and hidden layer nodes. Similar to this relation weight matrix can be specified as W_2 between hidden layer nodes and output nodes. For hidden layer and output layer nodes, threshold matrix is θ_1 and θ_2 , respectively:

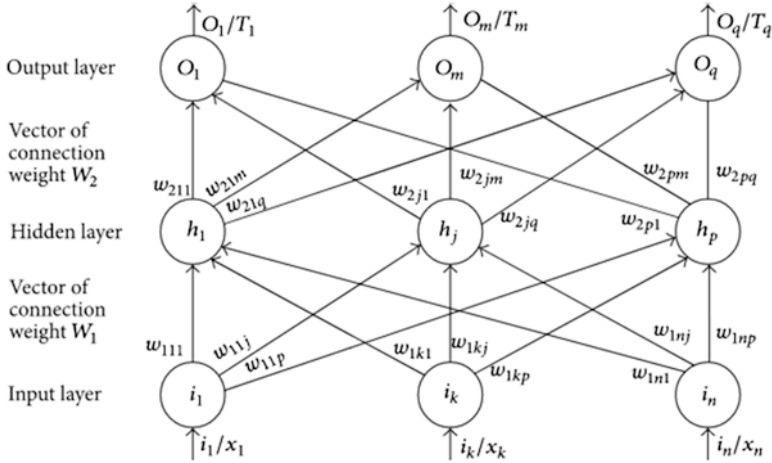


Fig. 3.3 Topology structure of BPNN

$$W_1 = \begin{cases} w_{111} & \dots & w_{11j} & \dots & w_{11p} \\ \vdots & & \vdots & & \vdots \\ w_{1k1} & \dots & w_{1kj} & \dots & w_{1kp} \\ \vdots & & \vdots & & \vdots \\ w_{1n1} & \dots & w_{1nj} & \dots & w_{1np} \end{cases} \quad (3.1)$$

3.3.4.2 Learning Algorithm of BP Network

BP network learning method is split into 2 steps which are propagated forward and back. It is a gradient descent method which can make error of per connection weights of NN reduce. Assign random numbers in $[-1,+1]$ set to relation weight matrices W_1 , W_2 and threshold matrices of hidden layer and output layer nodes θ_1 and θ_2 at beginning of learning.

3.3.4.3 Forward Operation

Next, enter $\{X, T\}$ learning samples, where X is learning samples' input vector and T is corresponding output vector:

$$X = (x_1, x_2, \dots, x_k, \dots, x_n), 1 \leq k \leq n \quad (3.2)$$

$$T = (T_1, T_2 \dots T_m \dots T_q), 1 \leq m \leq q \quad (3.3)$$

N is number of input layer nodes; q is number of output layer nodes.

Forward data propagates through input layer and hidden layer to output layer. Created weight value is learning outcome of classification of performance patterns. They primarily contain following measure.

Step 1: (Output value estimation of hidden layer nodes). Nodes value in hidden layer is

$$i_{1j} = \sum_{k=1}^n w_{1kj} x_k, 1 \leq j \leq p \quad (3.4)$$

If n is number of nodes of input layer, p is number of nodes of hidden layer, w_{1kj} is relationship weight, and x_k is input vector component.

Nodes output value is

$$h_j = f(i_{1j} + \theta_{1j}) \quad (3.5)$$

Where threshold of node j is θ_{1j} .

Sigmoid function given by Rumelhart is activation function f used:

$$f(i_{1j}) = \frac{1}{1 + e^{-i_{1j}}} \quad (3.6)$$

Step 2: (Calculating output value of output layer nodes). Input values for output layer nodes m are

$$i_{2m} = \sum_{j=1}^p w_{2jm} h_j, 1 \leq m \leq j \quad (3.7)$$

Node m output value is

$$O_m = f(i_{2m} + \theta_{2m}) \quad (3.8)$$

Where θ_{2m} is output layer node threshold m and f is activation function specified by sort (3.5).

3.3.4.4 Back Operation

Evaluate output value error and predicted output layer value error. From output layer, back propagation error via hidden to input layer. Values of connection weight are adjusted. Steps are as follows.

Step 1: (Calculation of output error of output layer nodes). Error between O_m learning value of output layer m node and output value of T_m learning samples is

$$\varepsilon_m = |O_m - T_m| \quad (3.9)$$

Step 2: (learning error testing). ε_0 is maximum learning error, which is set by user in $[0,1]$ interval.

When $\max(\varepsilon_m) \leq \varepsilon_0$, enter next learning sample. Network weights are adjusted and original learning samples are reentered. When all learning samples fulfill above criteria, learning process is terminated.

Step 3: (evaluate learning error in output layer nodes). Nodes output layer m, learning error is

$$d_{2m} = O_m (1 - O_m) (O_m - T_m) \quad (3.10)$$

Step 4: (evaluate learning error of hidden layer nodes). Hidden layer j learning error is

$$d_{1j} = h_j (1 - h_j) \sum_{m=1}^q w_{2jm} d_{2m} \quad (3.11)$$

Step 5 (revise value of connection weights matrix w_2). Set weight value as modified current weight value at time $t+1$; then set new weight value.

$$w_{2jm}(t+1) = w_{2jm}(t) + \eta d_{2m} h_j + \alpha [w_{2jm}(t) - w_{2jm}(t-1)] \quad (3.12)$$

where learning rate is η and momentum factor is α . In scope $[0,1]$, both α and η exists. Using alpha will speed up learning speed and help to solve common BP algorithm's local minimum problem.

Step 6: (revise value of connection weight matrix w_2). Consider,

$$w_{1kj}(t+1) = w_{1kj}(t) + \eta d_{1j} x_k + \alpha [w_{1kj}(t) - w_{1kj}(t-1)] \quad (3.13)$$

Step 7: (revise threshold θ_2). Output node layer threshold is

$$\theta_{2m}(t+1) = \theta_{2m}(t) + \eta d_{2m} h_j + \alpha [w_{2jm}(t) - w_{2jm}(t-1)] \quad (3.14)$$

Step 8: (revise threshold θ_1). Hidden layer threshold layer is

$$\theta_{1j}(t+1) = \theta_{1j}(t) + \eta d_{1j} + \alpha [\theta_{1j}(t) - \theta_{1j}(t-1)] \quad (3.15)$$

3.3.4.5 Run the BP Neural Network

Following learning, run BPNN and carry out pattern recognition for input vector I. BP learning algorithm's forward operation is used. Steps are as follows.

Step 1: Assign BP learning algorithm's updated value to relation weight matrices w_1 and w_2 and threshold matrices θ_1 and θ_2 .

Step 2: Input input vector I that requires recognized.

Step 3: To measure output of hidden layer, apply formulas (3.3), (3.4), and (3.5).

Step 4: To measure output of output layer, that is, classification result of input vector I, use formulas (3.5), (3.6), and (3.7).

3.3.5 Proposed Classification Using SLFN-COVID

Several attempts are made to overcome the flaws of Back Propagation Artificial Neural Network (BPANN). It revealed that SLFN with a maximum of hidden nodes was able to evaluate function for m various vectors present in the dataset used for training.

For m samples in $\mathbf{D} = \{(\mathbf{x}^{(k)}, \mathbf{t}^{(k)}) \mid \mathbf{x}^{(k)} \in \mathbf{R}^n, \mathbf{t}^{(k)} \in \mathbf{R}^p, k = 1, \dots, m\}$ where $\mathbf{x}^{(k)} = [x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}]^T$ and $\mathbf{t}^{(k)} = [t_1^{(k)}, t_2^{(k)}, \dots, t_p^{(k)}]^T$ represents features and target respectively in the training dataset. SLFN having M hidden nodes with activation function $g(x)$ as well as linear activation function in output nodes it is shown in Eq. 3.16 as:

$$\sum_{i=1}^M \beta_i g_i(\mathbf{x}^{(k)}) = \sum_{i=1}^M \beta_i g(w_i \cdot \mathbf{x}^{(k)} + b_i) = \mathbf{o}^{(k)}, \mathbf{K} = 1, \dots, m \quad (3.16)$$

Where, $\mathbf{w}_i \in \mathbf{R}^n$ gives the weights of input and i th hidden nodes in Eq. 3.17

$$\mathbf{w}_i = [w_{i1}, w_{i2}, \dots, w_{in}]^T, \quad (3.17)$$

$\beta_i \in \mathbf{R}^p$ is the weights of i^{th} hidden and output nodes in Eq. 3.18

$$\beta_i = [\beta_{i1}, \beta_{i2}, \dots, \beta_{ip}]^T, \quad (3.18)$$

$\mathbf{w}_i \bullet \mathbf{x}^{(k)}, b_i, \mathbf{o}^{(k)} \in \mathbf{R}^p$ represents the inner product of \mathbf{w}_i and $\mathbf{x}^{(k)}$, bias of i th hidden node and output of NN for k th vector respectively. SLFN estimates m vectors i.e., \mathbf{w}_i, β_i , and b_i exists such that in Eqs. 3.19 and 3.20:

$$\mathbf{o}^{(k)} - \mathbf{t}^{(k)} = \mathbf{0} \quad (3.19)$$

$$\sum_{i=1}^M \beta_i g(w_i \cdot \mathbf{x}^{(k)} + b_i) = \mathbf{t}^{(k)}, k = 1, \dots, m \quad (3.20)$$

This equation can also be represented as: $H\beta = T$

where, $H \in \mathbb{R}_{m \times M}$ is hidden layer output matrix of NNs.

$$H = \begin{bmatrix} g(w_1 \cdot x^{(1)} + b_1) & \cdots & g(w_M \cdot x^{(1)} + b_M) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x^{(m)} + b_1) & \cdots & g(w_M \cdot x^{(m)} + b_M) \end{bmatrix} \quad (3.21)$$

$\beta \in \mathbb{R}_{M \times p}$ specifies weights of output and hidden layers.

$$\beta = \begin{bmatrix} \beta_1^T \\ \cdot \\ \cdot \\ \beta_M^T \end{bmatrix} \quad (3.22)$$

$T \in \mathbb{R}_{m \times p}$ represents target values of m vectors in training dataset

$$T = \begin{bmatrix} t^{(1)T} \\ \cdot \\ \cdot \\ \cdot \\ t^{(m)T} \end{bmatrix} \quad (3.23)$$

In each iteration of conventional gradient descent based learning approach, the weights w_i that connected input and hidden layers and biases b_i has to be initialized and adjusted. This was the main reason for making the training process often time consuming and hence the training model at times fails to obtain global minima. But, the minimum norm least-squares used in SLFN avoided tuning those parameters. In the training process in Eq. 3.24, SLFN with fixed w_i and b_i was identical for estimating the least square solution $\hat{\beta}$ of $H\beta = T$

$$H(w_1, \dots, w_M, b_1, \dots, b_M) \hat{\beta} - T = \min_{\beta} H(w_1, \dots, w_M, b_1, \dots, b_M) \beta - T \quad (3.24)$$

with the smallest value $\hat{\beta} = H^\dagger T$

where, H^\dagger was Moore-Penrose generalized inverse of matrix H. Basic significant properties of this solution are less training error, unique solution and smallest weight norms. Least norm least-square solution provided for SLFN is termed as ELM which involves the following steps during training process:

- w_i and b_i , for $i = 1, \dots, M$ were set randomly.
- H , the matrix of the hidden layer output, was computed
- $\hat{\beta}$ was determined

To conclude, three major differences between BPANN and ELM-ANN were identified which are as follows:

- BP-ANN has to tune parameters like learning rates, momentum, number of hidden nodes, and terminating norm. But, ELM-ANN avoid these operations and defines only the number of hidden nodes.
- It is operated on differentiable activation functions only whereas ELM-ANN uses both type of functions.
- Both BP-ANN and ELM-ANN obtain their trained model with minimum training error. The difference is that the former possibly reaches local minima while the latter additionally obtains smallest norm of weight thereby providing better improved generalized model and reaches global minima.

3.3.5.1 Dataset

COVID-19 was declared as Health Emergency by WHO and thus hospitals and researchers gave open access to the data related to this virus. The dataset employed for an analysis in this study was developed by Hannah Ritchie [30] which is regularly updated with the reports of WHO [31]. The additional information related to this dataset can be obtained from <https://ourworldindata.org/coronavirus-source-data>.

3.4 Experimental Analysis

The parameters used for analysis are Accuracy, F1 score, recall and precision. Notations TP, FN, TN and FP represented in equations denote True Positive, False Negative, True Negative and False Positive.

Accuracy This is ratio of true predictions to total data predicted by the classifier. This significant measure is used to evaluate the classification model is given by in Eq. 3.25:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.25)$$

Precision This is ratio of samples that are aptly identified to the sum of identified samples which is given below in Eq. 3.26:

$$Precision = \frac{TP}{TP + FP} \quad (3.26)$$

Recall This is ratio of correctly identified samples to sum of samples that are TP and FN which is shown in Eq. 3.27:

$$Recall = \frac{TP}{TP + FN} \tag{3.27}$$

F1 Score It provides a perfect evaluation of classifying model which is computed in Eq. 3.28:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.28}$$

The Table 3.1 analyzes the accuracy of existing and proposed methods.

Figure 3.4 illustrates the comparative analysis of accuracy for existing and proposed methods. X axis and Y axis provides number of samples and Accuracy in

Table 3.1 Accuracy analysis

No. of samples	ML based improved model [7]	SPP-COVID-Net classifier [10]	SLFN-COVID classifier
100	54	60	67
200	61	69	78
300	67	76	84
400	78	84	89
500	86	89	95

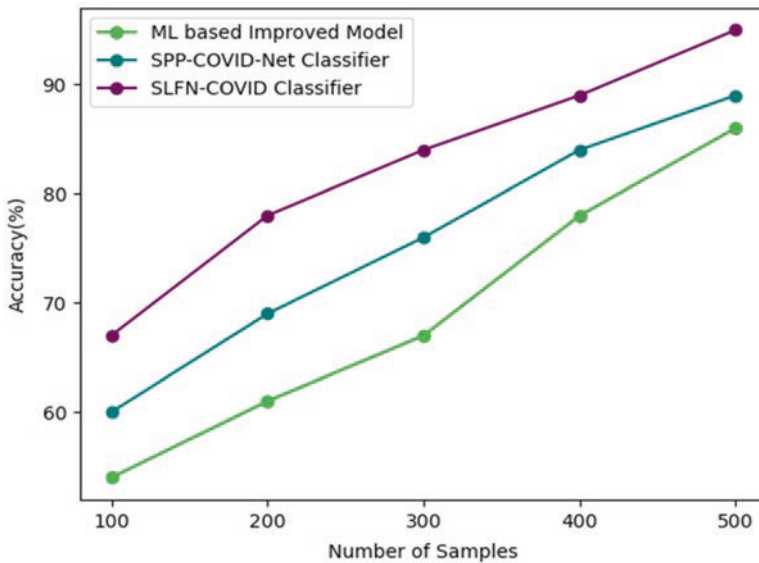


Fig. 3.4 Comparison of accuracy

percentage respectively. Green, teal and violet color indicates ML based Improved Model, SPP-COVID-Net classifier and SLFN-COVID Classifier respectively. The proposed method achieves 83% of accuracy. The Table 3.2 shows analysis of precision with existing and proposed techniques.

Figure 3.5 presents comparative analysis of precision for existing and proposed methods. X axis and Y axis provides number of samples and precision in percentage respectively. Green, teal and violet color indicates ML based Improved Model, SPP-COVID-Net classifier and SLFN-COVID Classifier respectively. The proposed method achieves 73% of precision. Table 3.3 analyzes recall of the existing and proposed methods.

Figure 3.6 illustrates comparative analysis of recall for existing and proposed methods. X axis and Y axis provides number of samples and recall in percentage respectively. Green, teal and violet color indicates ML based Improved Model, SPP-COVID-Net classifier and SLFN-COVID Classifier respectively. When compared to existing methods, proposed method achieves 68% of recall. The Table 3.4 shows analysis of F1-Score with existing and proposed techniques.

Table 3.2 Precision analysis

No. of samples	ML based improved model [7]	SPP-COVID-Net classifier [10]	SLFN-COVID classifier
100	47	51	59
200	51	62	68
300	58	68	71
400	65	70	79
500	71	75	89

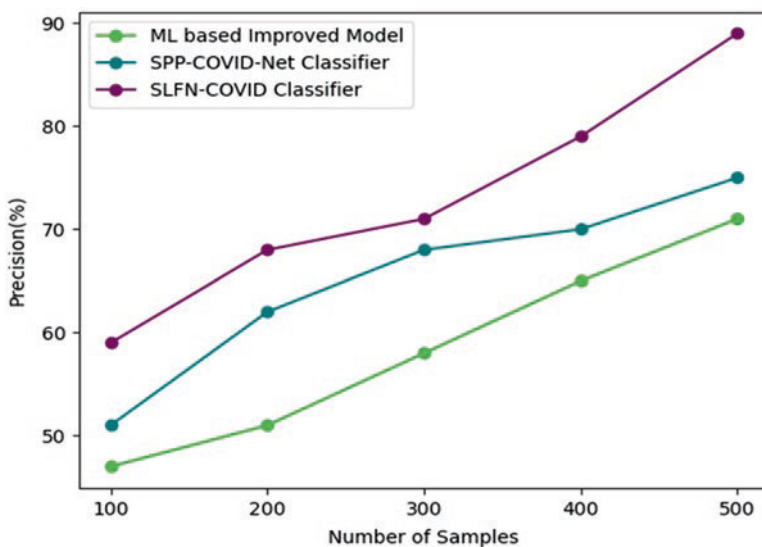


Fig. 3.5 Comparison of precision

Table 3.3 Recall analysis

No. of samples	ML based improved model [7]	SPP-COVID-Net classifier [10]	SLFN-COVID classifier
100	38	39	56
200	41	47	65
300	49	51	69
400	56	60	72
500	61	68	78

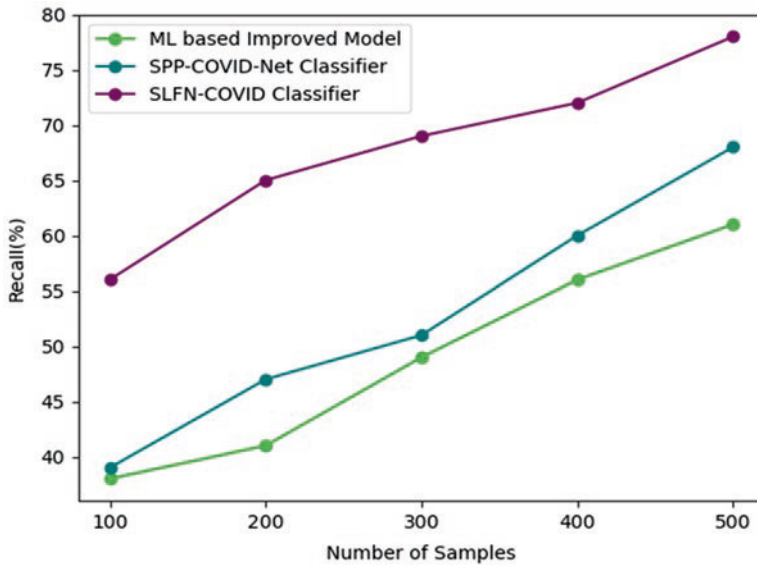


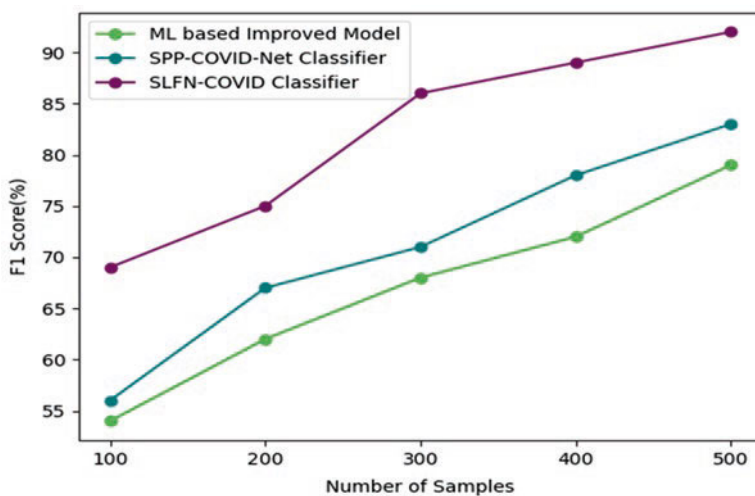
Fig. 3.6 Comparison of recall

Figure 3.7 compares F1 score obtained for existing and proposed methods. X axis and Y axis provides number of samples and F1-Score in percentage respectively. Green, teal and violet color indicates ML based Improved Model, SPP-COVID-Net classifier and SLFN-COVID Classifier respectively. Proposed method achieves 82% of F1-Score. Table 3.5 presents overall performance of existing and proposed techniques.

Figure 3.8 compares overall performance of existing and proposed methods. X axis provides parameters used for analysis and their corresponding values in percentage in Y axis. The pink, lavender and violet color indicates ML based Improved Model, SPP-COVID-Net classifier and SLFN-COVID Classifier respectively. When compared to the existing methods, proposed method achieves good results.

Table 3.4 F1-score analysis

No. of samples	ML based improved model [7]	SPP-COVID-Net classifier [10]	SLFN-COVID classifier
100	54	56	69
200	62	67	75
300	68	71	86
400	72	78	89
500	79	83	92

**Fig. 3.7** Comparison of F1-score**Table 3.5** Overall Performance of the existing and proposed methods

Parameters	ML based improved model [7]	SPP-COVID-Net classifier [10]	SLFN-COVID classifier
Accuracy	69%	76%	83%
Precision	58%	65%	73%
Recall	49%	53%	68%
F1-Score	67%	71%	82%

3.5 Conclusion

Several approaches like GAN, RNN, LSTM, and ELM are developed for diagnosing COVID-19. The major issues related to COVID-19 are high-risk people, geographical issues, and radiology and recognizing which have been analyzed and described in this work. The proposed SLFN-COVID Classification method is used

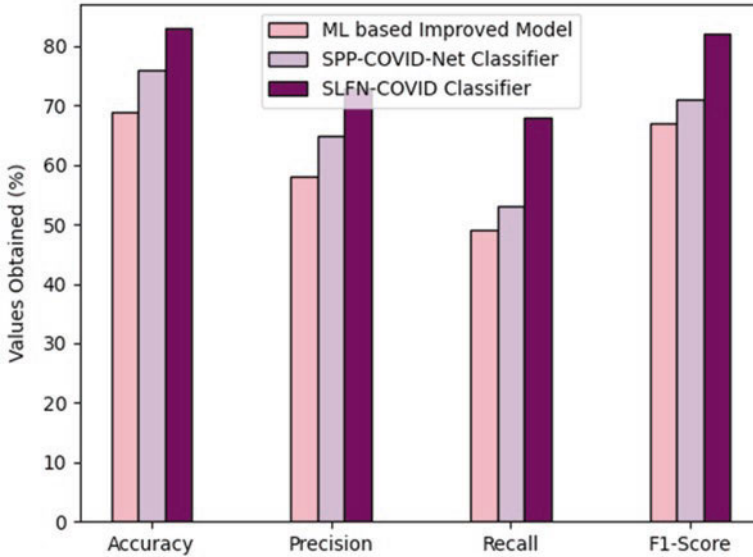


Fig. 3.8 Overall comparison of existing and proposed methods

for detection and diagnosing the disease based on the patient COVID-19 dataset which create the social impacts and its helps for treatment. The experimental results achieved are 83% of accuracy, 73% of precision, 68% of recall, and 82% of F1-Score. The SLFN-COVID classifier exceeds prior approaches. It is concluded that the the future effort is to focus on the growth of health services which led to an increase in data generation causing problems in the design of data access and data structures. An additional effort must therefore be undertaken to address the complexity and diversity of healthcare system data. In addition, a uniform standard of heterogeneous data management is necessary because of the diversity of data sources; for the diversity of the data content, a single programing interface is required for several data analytics modules.

References

1. Mukherjee, A., Ghosh, S., Behere, A., Ghosh, S. K., & Buyya, R. (2021). Internet of health things (IoHT) for personalized health care using integrated edge-fog-cloud network. *Journal of Ambient Intelligence and Humanized Computing*, 12, 943–959.
2. Punn, N. S., Sonbhadra, S. K., & Agarwal, S. (2020). COVID-19 epidemic analysis using machine learning and deep learning algorithms. MedRxiv.
3. Medscape Medical News, The WHO declares public health emergency for novel coronavirus. (2020). <https://www.medscape.com/viewarticle/924596>

4. Sampathkumar, A., Murugan, S., Rastogi, R., Mishra, M. K., Malathy, S., & Manikandan, R. (2020). Energy efficient ACPI and JEHD0 mechanism for IoT device energy management in healthcare. In *Internet of things in smart technologies for sustainable urban development* (pp. 131–140).
5. Khanday, A. M. U. D., Rabani, S. T., Khan, Q. R., Rouf, N., & Din, M. M. U. (2020). Machine learning based approaches for detecting COVID-19 using clinical text data. *International Journal of Information Technology*, 12(3), 731–739.
6. Iwendi, C., Bashir, A. K., Peshkar, A., Sujatha, R., Chatterjee, J. M., Pasupuleti, S., & Jo, O. (2020). COVID-19 patient health prediction using boosted random forest algorithm. *Frontiers in Public Health*, 8, 357.
7. Tuli, S., Tuli, S., Tuli, R., & Gill, S. S. (2020). Predicting the growth and trend of COVID-19 pandemic using machine learning and cloud computing. *Internet of Things*, 11, 100222.
8. Sujath, R., Chatterjee, J. M., & Hassanien, A. E. (2020). A machine learning forecasting model for COVID-19 pandemic in India. *Stochastic Environmental Research and Risk Assessment*, 34, 959–972.
9. Jiang, X., Coffee, M., Bari, A., Wang, J., Jiang, X., Huang, J., & Huang, Y. (2020). Towards an artificial intelligence framework for data-driven prediction of coronavirus clinical severity. *Computers, Materials & Continua*, 63(1), 537–551.
10. Abdani, S. R., Zulkifley, M. A., & Zulkifley, N. H. (2020). A lightweight deep learning model for covid-19 detection. In *2020 IEEE Symposium on Industrial Electronics & Applications (ISIEA)* (pp. 1–5).
11. He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9), 1904–1916.
12. Abdani, S. R., & Zulkifley, M. A. (2019). Densenet with spatial pyramid pooling for industrial oil palm plantation detection. In *2019 International conference on Mechatronics, Robotics and Systems Engineering (MoRSE)* (pp. 134–138).
13. Hossain, M. S., & Muhammad, G. (2016). Cloud-assisted industrial internet of things (iiot)–Enabled framework for health monitoring. *Computer Networks*, 101, 192–202.
14. Stojanović, O., Leugering, J., Pipa, G., Ghazzi, S., & Ullrich, A. (2019). A bayesian monte carlo approach for predicting the spread of infectious diseases. *PLoS one*, 14(12).
15. Singh, D., Kumar, V., & Kaur, M. (2020). Classification of COVID-19 patients from chest CT images using multi-objective differential evolution–based convolutional neural networks. *European Journal of Clinical Microbiology & Infectious Diseases*, 39(7), 1379–1389.
16. Gangavarapu, T., & Patil, N. (2019). A novel filter–wrapper hybrid greedy ensemble approach optimized using the genetic algorithm to reduce the dimensionality of high-dimensional biomedical datasets. *Applied Soft Computing*, 81, 105538.
17. Zhou, L., Li, Z., Zhou, J., Li, H., Chen, Y., Huang, Y., & Gao, X. (2020). A rapid, accurate and machine-agnostic segmentation and quantification method for CT-based COVID-19 diagnosis. *IEEE Transactions on Medical Imaging*, 39(8), 2638–2652.
18. Arun, S. S., & Iyer, G. N. (2020). On the analysis of COVID19–Novel corona viral disease pandemic spread data using machine learning techniques. In *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 1222–1227).
19. Oh, Y., Park, S., & Ye, J. C. (2020). Deep learning covid-19 features on cxr using limited training data sets. *IEEE Transactions on Medical Imaging*, 39(8), 2688–2700.
20. Marmarelis, V. Z. (2020). Predictive modeling of Covid-19 data in the US: Adaptive phase-space approach. *IEEE Open Journal of Engineering in Medicine and Biology*, 1, 207–213.
21. Barabas, J., Zalman, R., & Kochlán, M. (2020). Automated evaluation of COVID-19 risk factors coupled with real-time, indoor, personal localization data for potential disease identification, prevention and smart quarantining. In *2020 43rd International conference on Telecommunications and Signal Processing (TSP)* (pp. 645–648).
22. Fan, D. P., Zhou, T., Ji, G. P., Zhou, Y., Chen, G., Fu, H., & Shao, L. (2020). Inf-net: Automatic covid-19 lung infection segmentation from ct images. *IEEE Transactions on Medical Imaging*, 39(8), 2626–2637.

23. Ye, Y., Hou, S., Fan, Y., Qian, Y., Zhang, Y., Sun, S., & Laparo, K. (2020). α -Satellite: An AI-driven system and benchmark datasets for hierarchical community-level risk assessment to help combat COVID-19. arXiv preprint arXiv:2003.12232.
24. Gaur, L., Bhatia, U., Jhanjhi, N. Z., Muhammad, G., & Masud, M. (2021). Medical image-based detection of COVID-19 using Deep Convolution Neural Networks. In *Multimedia Systems* (Vol. 1–10).
25. Wang, Y., Niu, D., & Ji, L. (2012). Short-term power load forecasting based on IVL-BP neural network technology. *Systems Engineering Procedia*, 4, 168–174.
26. You, X., & Cao, X. (2015). Study of liquid lithium coolant interaction based on BP neural network optimized by genetic algorithm. *Journal of Fusion Energy*, 34(4), 918–924.
27. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533–536.
28. Xie, Z., Du, Q., Ren, F., Zhang, X., & Jamiesone, S. (2015). Improving the forecast precision of river stage spatial and temporal distribution using drain pipeline knowledge coupled with BP artificial neural networks: A case study of Panlong River, Kunming, China. *Natural Hazards*, 77(2), 1081–1102.
29. Wang, S. T., Chen, D. Y., Hou, P. G., Wang, X. L., Wang, Z. F., & Wei, M. (2015). Determination of the sodium methylparaben content based on spectrum fluorescence spectral technology and GA-BP neural network. *Guang pu xue yu guang pu fen xi= Guang pu*, 35(6), 1606–1610.
30. Our World In Data: COVID-19 Dataset; source: <https://github.com/owid/covid-19-data/tree/master/public/data/>
31. Situation Reports-WHO; <https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports>