Article

Detecting COVID-19 Status Using Chest X-ray Images and Symptoms Analysis by Own Developed Mathematical Model: A Model Development and Analysis Approach

Mohammad Helal Uddin †, Mohammad Nahid Hossain †,*, Md Shafiqul Islam, Md Abdullah Al Zubaer and Sung-Hyun Yang *

Electronic Engineering Department, Kwangwoon University, Seoul 139-701, Korea;
helaluddinsimon@kw.ac.kr (M.H.U.); msislam@kw.ac.kr (M.S.I.); abdullah@kw.ac.kr (M.A.A.Z.)
* Correspondence: nahid@kw.ac.kr (M.N.H.); shyang@kw.ac.kr (S.-H.Y.)
† These authors contributed equally to this work.

Abstract: COVID-19 is a life-threatening infectious disease that has become a pandemic. The virus grows within the lower respiratory tract, where early-stage symptoms (such as cough, fever, and sore throat) develop, and then it causes a lung infection (pneumonia). This paper proposes a new artificial testing methodology to determine whether a patient has been infected by COVID-19. We have presented a prediction model based on a convolutional neural network (CNN) and our own developed mathematical equation-based algorithm named SymptomNet. The CNN algorithm classifies lung infections (pneumonia) using frontal chest X-ray images, and the symptom analysis algorithm (SymptomNet) predicts the possibility of COVID-19 infection from the developed symptoms in a patient. By combining the CNN image classifier method and SymptomNet algorithm, we have developed a model that predicts COVID-19 patients with an approximate accuracy of 96%. Ten out of the 13 symptoms were significantly correlated to the COVID-19 disease. Specially, fever, cough, body chills, shortness of breath, muscle pain, and sore throat were shown to be significantly related ($r = 0.20$; $p = 0.001$, $r = 0.20$; $p < 0.001$, $r = 0.22$; $p < 0.001$, $r = 0.16$; $p < 0.001$, $r = -0.45$; $p < 0.001$, $r = -0.35$; $p < 0.001$, respectively). In this model, the CNN classifier has an accuracy of approximately 96% (training loss = 0.1311, training accuracy = 0.9596, validation loss: 0.2754, and validation accuracy of 0.9273, F1-score: 94.16, precision: 91.33), and the SymptomNet algorithm has an accuracy of 97% (485 successful predictions out of 500 samples). This research work obtained promising accuracy while predicting COVID-19-infected patients. The proposed model can be ubiquitously used at a low cost and achieve high accuracy.

Keywords: COVID-19; machine learning; convolutional neural network; lung infection (pneumonia); frontal chest X-ray; symptom analysis

1. Introduction

The new coronavirus carried by bats called COVID-19 has become a pandemic and has created a dreadful situation worldwide. The virus became dangerous at the end of November 2019 when it underwent a mutation, evolving into the viral strain that we now call SARS-CoV-2 [1]. According to the WHO, there is no alternative to detect this disease other than testing [2]. We have developed a solution that combines deep learning and our own proprietary mathematical equation-based algorithm named SymptomNet. This method might help to accelerate the mission of predicting and identifying possible COVID-19 patients. This study proposed a model to predict COVID-19 patients in the early stages by applying a deep learning-based Convolutional Neural Network (CNN) algorithm to frontal chest X-ray images, which is coupled with our own proprietary symptom analysis algorithm (SymptomNet). According to a study from the UK's King's

College London, there could be six distinct types of COVID-19, and each type could be distinguished by its own cluster of symptoms [3]. Therefore, we designed our algorithm in such a way that if the virus mutates, our algorithm will still be able to predict COVID-19 patients with 97% accuracy.

Some practical analyses, reports, and articles [4–6] on COVID-19 are available. The characteristics [7] of COVID-19 have been analyzed for different types of patients [8]. Some phone-based surveys [9] of populations when cities and towns were under quarantine were conducted, and the results have been analyzed using an artificial intelligence (AI) framework. Reports on the sensitivity of chest [10] COVID-19 pathological findings [6] associated with acute respiratory distress syndrome have been published. Comparatively, the reproductive rate of COVID-19 being higher than that of the SARS coronavirus [11] has been discussed. Some mobile applications, such as "WASHKARO" [12] have been developed to raise public awareness of the initial safety instructions [13] of the "World Health Organization (WHO)" to fight the novel coronavirus disease COVID-19 in the initial stages.

One of the most important parts of this study was the detection of an infected lung (pneumonia by COVID-19) since this is a vital symptom of COVID-19. Pneumonia or pneumonitis is a fiery condition that principally influences the lungs. To evaluate the seriousness of COVID-19 pneumonia, chest CTs of the lungs were used. Basically, this method determines COVID-19 pneumonia from the initial diagnosis associated with the change in chest CT findings until the patient recovers [14]. A survey provided a reliable estimate of approximately 1 million people being hospitalized in the United States [15] because of pneumonia-related conditions per year. A team was formed in the USA to identify and classify pneumonia cases. Features from images are extracted utilizing diverse neural network models pre-trained using ImageNet, which are then fed into a classifier for prediction. This method achieved a precision of 96.4% and a recall of 99.62% [16]. The k-nearest neighbors, linear discriminant analysis, linear regression, and support vector machine (SVM) learning models were used to detect pneumonia quickly with 99.41% accuracy [17]."CheXnet" with a "121-layer convolutional neural network" trained on ChestX-ray14, which claims to be the largest publicly accessible chest X-ray dataset, containing more than 100K frontal view X-ray images of 14 diseases, can recognize pneumonia from chest X-rays at a level surpassing experienced radiologists with 85% accuracy [18]. Furthermore, DCNNs [19–21] use constructive work combined with frontal chest X-rays to determine whether a lung is infected (pneumonia) or not and perform symptom analysis, achieving 96% accuracy on a large-scale radiology database [22]. Symptom analysis is the first requirement to diagnose flu or other diseases. Natural language processing (NLP) is used for symptom analysis of patients [23]. A study that analyzes the clinical characteristics and identifies the predictors of disease severity and mortality based on the recent "Coronavirus Disease 2019 (COVID-19)" outbreak started in Wuhan, China has been conducted in the symptom analysis field. They summarize the clinical features of the confirmed COVID-19 patients and further identify the risk factors for disease severity and death [24].

This study combines two steps to identify COVID-19. The first step involves developing a CNN-based classifier to identify normal or infected lungs (pneumonia) from frontal chest X-ray images, and the second step analyzes the symptoms of COVID-19 using our own proprietary algorithm, SymptomNet, to predict infections. Our proposed method highlights the following contributions:

(1)   We have proposed a composition of chest X-ray and symptoms data to detect COVID-19 status.

(2)   A formulaic solution has been proposed in this regard.

(3)   We have proposed the SymptomNet algorithm.

The system has been cross-validated with real-time survey data collected from newspapers, social media posts, TV interviews, relatives of patients, mobile surveys, etc. This would be a naive method to test people all around the world to alleviate coronavirus

transmission with our system, which has an attractive accuracy rate of approximately 96%.

2. Materials and Methods

A deep learning-based convolutional neural network (CNN) has been used, and the CNN consists of 7 layers: one input layer; 5 hidden layers, which are the ConV 2D (3 × 3, 16), ConV 2D (3 × 3, 32), ConV 2D (3 × 3, 64), ConV 2D (3 × 3, 128), and ConV 2D (3 × 3, 256) layers, where 3 × 3 is the kernel size, 16, 32, 64, 128, and 256 are the respective filter sizes and 2×2 is the pooling size with the ReLU activation function, and one FC (Fully Connected) layer. The frontal chest X-ray images are used as the input images. They pass through filters, the ReLU activation function, and the sigmoid activation function to increase the nonlinearity. After this pooling layer is applied to each feature map and the pooled images are flattened into one long vector, then this vector will be the input of a fully connected artificial neural network. Through these preprocessing steps, the system will be trained by forward- and backpropagation for a large number of epochs. After several steps, the system can identify normal or infected lungs from X-ray images (pneumonia). The "Adam optimizer" algorithm has been used to update the weight parameters to optimize the loss function. The following is a summary of the main model; the details will be described in the methodology.

In the symptom analysis step, the SymptomNet algorithm with two mathematical linear equations is developed to analyze the symptoms. The first equation helps to predict the status of the COVID-19 patient based on the symptoms that they have developed. The other equation is derived to set the threshold point for the first equation. Then, the algorithm predicts whether a patient has COVID-19 or not. The details will be described in the methodology section.

After obtaining the results from the CNN algorithm and SymptomNet algorithm, this study will predict whether a person has COVID-19. The final symptom analysis test was performed on a dataset of 500 COVID-19-positive patients, which was collected manually in Bangladesh. Among these 500 patients, we also collected 10 patients' frontal chest X-ray images to test our final model.

To conduct the image processing, we used the Keras deep learning framework with the TensorFlow backend [25]. We ran all the experiments on a laboratory standard PC with an Nvidia GeForce GTX 1080 GPU card with 8 GB GDDR5X memory.

2.1. Proposed CNN Model

The CNN model combines an input layer and an output layer with hidden layers between them. The hidden layers normally comprise convolutional layers, ReLU layers, pooling layers, and fully connected layers. A classic CNN architecture would look something like the following Figure 1.
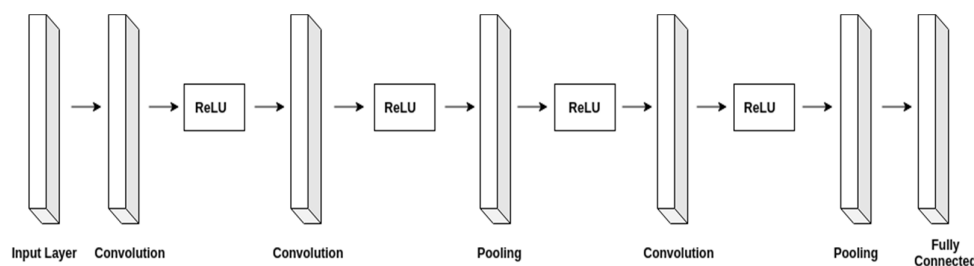

Figure 1. A classic CNN architecture.

Figure 2 shows the implemented CNN model architecture, which combines two main parts: the feature extractors and the classifier (sigmoid activation function). In the feature extraction layers, the input that has been taken is its immediately preceding layer's output,

and the output of the layer is passed to the succeeding layers as the input. The CNN architecture in Figure 2 is combined with the convolution, max pooling, and classification layers. We have used five convolutions and a fully connected layer between the input layer and output layer. The feature extractors include a conv2D with a 3 × 3 kernel size and 16 filters, a conv2D with a 3 × 3 kernel size and 32 filters, a conv2D with a 3 × 3 kernel size and 64 filters, a conv2D with a 3 × 3 kernel size and 128 filters, and a conv2D with a 3 × 3 kernel size and 256 filters 256; and there are ReLU activators between them.
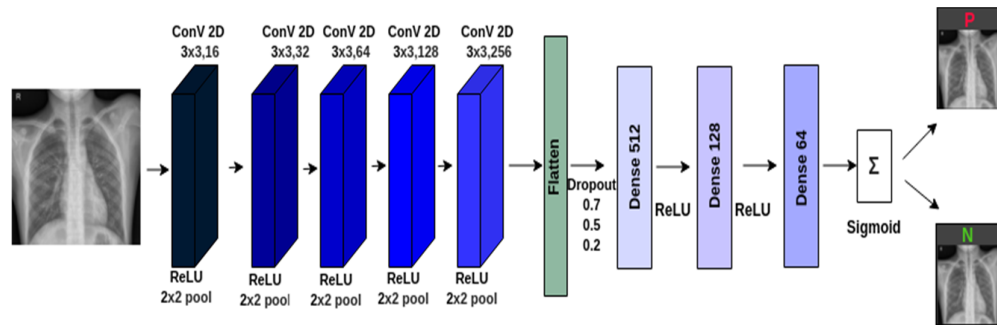

Figure 2. The proposed architecture of CNN.

2.2. Symptom Analysis Using the Proposed SymptomNet Algorithm

Based on 53,000 patient data samples [24], 12 symptoms have been found that are mainly responsible for COVID-19 identification. Fever, cough, fatigue, shortness of breath, and muscle pain are the top five symptoms of COVID-19. Chill, dizziness, headache, sore throat, nausea or vomiting, diarrhea, and nasal congestion can also when a person is infected by COVID-19. Figure 3 provides an overview of the COVID-19 symptoms.
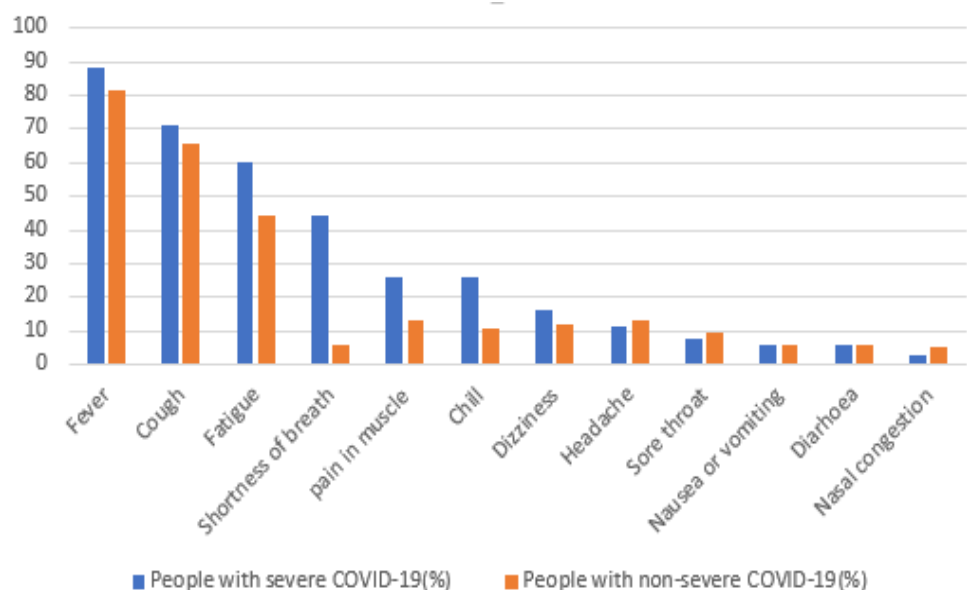

Figure 3. People with severe and non-severe COVID-19.

The analysis of the data sample shows that among the severe COVID-19 patients, 88.4% experience a fever, 71.1% experience coughing, 60.3% experience fatigue, 44.2% experience shortness of breath, 26% experience muscle pain, 26% experience chills, 16.1% experience dizziness, 11.3% experience headaches, 7.8% experience a sore throat, 5.9%

experience nausea or vomiting, 5.7% experience diarrhea, and 2.8% experience nasal congestion [24].

To predict COVID-19 based on symptoms, we have developed an algorithm with two mathematical linear equations. We have named our algorithm "SymptomNet". The equations are based on the symptom weights. Since all symptoms do not have the same impact on COVID-19, we assigned weights to each individual symptom. Regarding the weights of the individual symptoms, we considered the impact percentage of individual symptoms on COVID-19 based on the collected dataset. Table 1 presents the weights of the symptoms.

Table 1. Symptom weight table.

Clinical Symptoms	Symptom Weights (%)
Fever	88.4
Cough	71.1
Fatigue	60.3
Shortness of breath	44.2
Muscle pain	26
Chill	26
Dizziness	16.1
Headache	11.3
Sore throat	7.8
Nausea or vomiting	5.9
Diarrhea	5.7
Nasal congestion	2.8

This is the default symptom weight table for the SymptomNet algorithm. However, this symptom weight table can be changed. If someone wants to impose a new symptom weight table instead of the default table, the SymptomNet algorithm can accommodate this. Considering the weights of symptoms, we have derived the prediction equations as follows:

$$COVID-19'prediction, C_{pr} = \frac{\sum (w_1 x_1 + w_2 x_2 + \cdots \ldots + w_n x_n) \times 100}{\Sigma(w_1 + w_2 + \cdots \ldots + w_n)} \quad (1)$$

Here, w is denoted as the weight of the symptom, x is denoted as the symptom.

Based on the data sample, we considered 12 symptoms for the equation, but the equation is derived for n_th number of symptoms so that it can fit an increased/decreased number of symptoms. For Equation (1), we have considered symptom (x) as being binary, either 1 or 0. When the symptom is positive, it is 1; and when the symptom is negative, it is 0. In Equation (2), the threshold point is represented. After that point, for Equation (1), we are able to conclude that a patient is possibly COVID-19 positive. To calculate the threshold, we considered the weights of the top three symptoms of COVID-19 from the symptom weight table, which means that symptom weight Table 1 will be sorted in descending order (large to small). Equation (2) is derived to determine the threshold point for Equation (1).

$$Threshold, T_r = \frac{\sum_{n=1}^{3} LARGE(w_1 x_1 + w_2 x_2 + \ldots \ldots + w_n x_n +) \times 100}{\Sigma(w_1 + w_2 + \ldots \ldots + w_n)} \quad (2)$$

T_r is denoted as the threshold.

Figure 4 shows the working process of the SymptomNet algorithm.

This system is designed in such a way that the "symptom weight table" needs to be integrated with the system once, and then it will be updated automatically as it is input. The input parameters for the algorithm are patients' symptoms. After taking the input, Equation (1) will calculate the predicted value (C_{pr}) using the input and symptom weights. In addition, the threshold (T_r) value will be calculated from the "symptom weight table".

If the predicted value equals or exceeds the threshold value, then the system will predict a COVID-19 infection; otherwise, it will predict that the person is not infected.

The symptom pattern of COVID-19 can be changed. As a result, the weight of the symptoms of COVID-19 might vary from the default symptom weight table in Table 1.
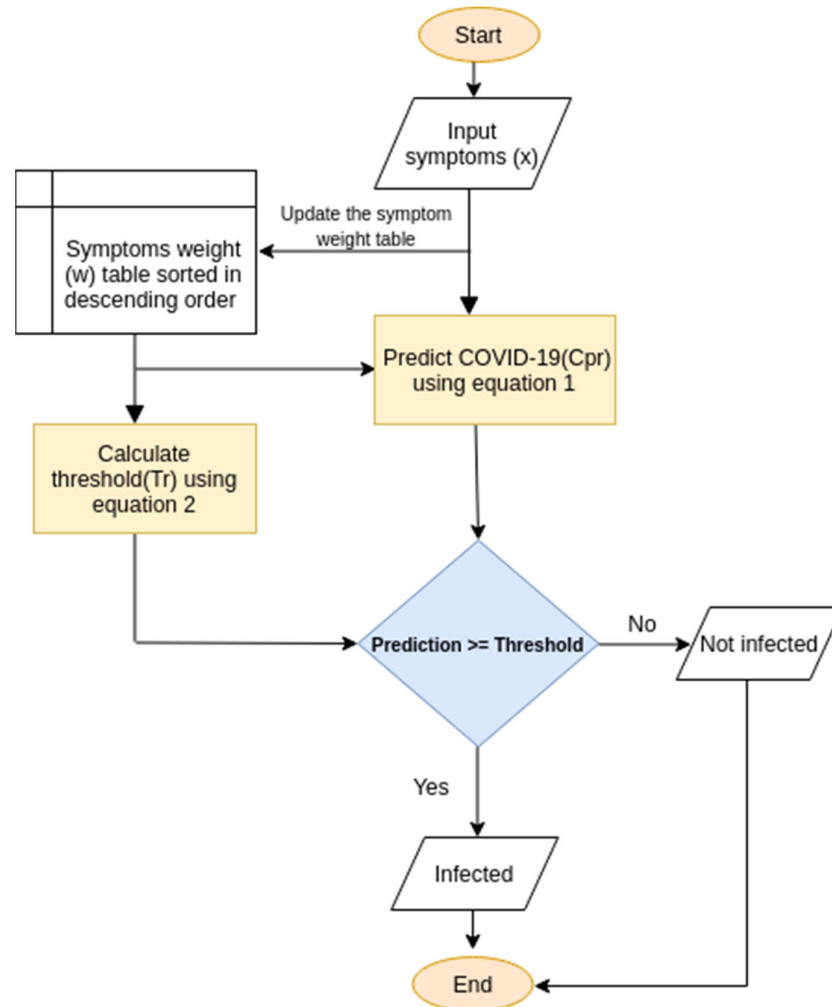

Figure 4. Flowchart of the symptom analysis algorithm (SymptomNet).

2.3. Combining the CNN and SymptomNet Algorithms

In this part, we have combined our CNN image classifier and the SymptomNet algorithm. In the previous two parts, we have individually described the CNN image classifier and the SymptomsNet algorithm in detail. Figure 5 displays the architecture of the full model of our work.

If both the CNN image classifier and the SymptomsNet algorithm indicate positive results for a patient, then we can conclude that there is a high possibility that the patient is infected by COVID-19 (Table 2).

If the CNN classifier indicates that patients have a lung infection, but the equation result is under the threshold, then we can conclude that there is a moderate possibility that the patient is infected by COVID-19. Furthermore, if the CNN classifier indicates a negative result for the lung infection, but the equation result is over the threshold, then we can conclude that there is a low possibility that the patient is infected by COVID-19.
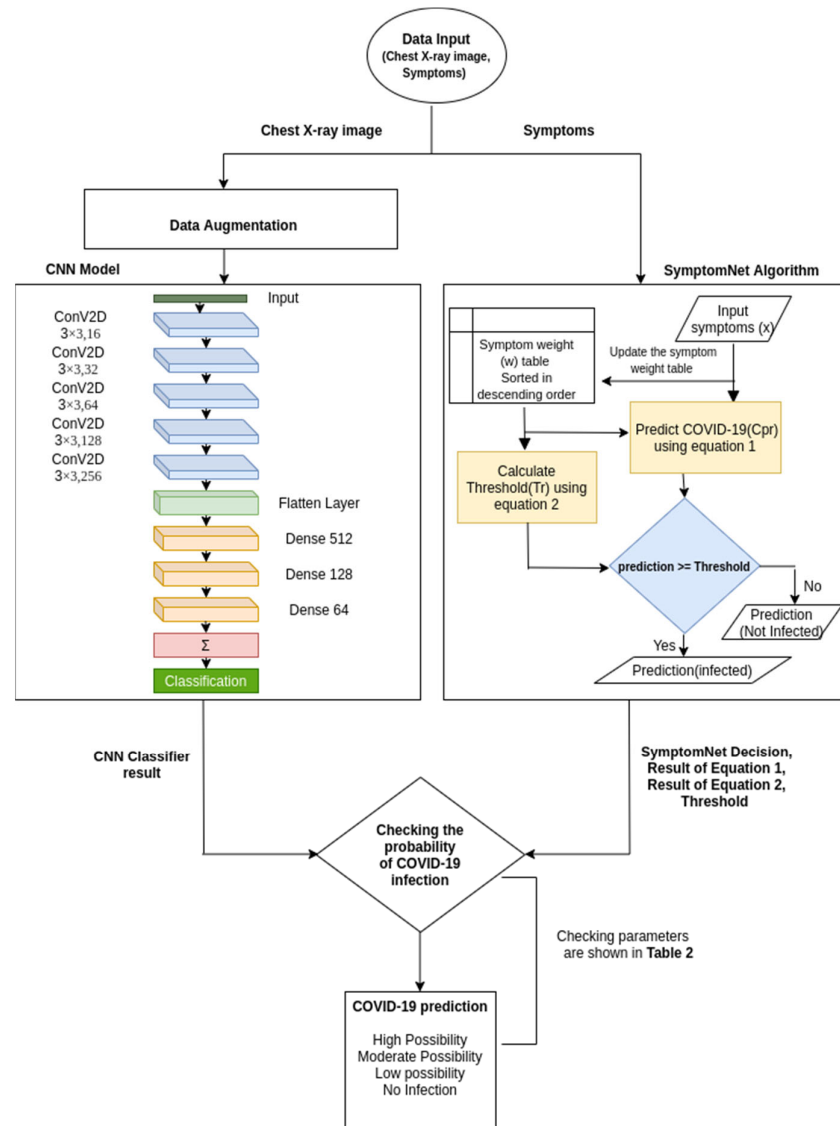
Figure 5. Research flowchart of the full proposed model.

Table 2. Conditions for possible COVID-19 status.

CNN Classifier	SymptomNet Algorithm	COVID-19 Prediction
Pneumonia	≥Threshold	High Possibility
Pneumonia	<Threshold	Moderate Possibility
Normal	Threshold < C_{pr} ≤ 70	Low possibility
Normal	C_{pr} > 70	High Possibility
Normal	<Threshold	No Infection

2.4. Experiments

2.4.1. Data Preprocessing and Augmentation

The dataset was released by Paul Mooney and is also publicly available on the Kaggle platform [26]. This dataset contains a total of 5862 frontal chest X-ray images from people of different ages and genders. This data set is divided into three subsets, which are test, training, and validation sets. We defined two data generators, one for the training data and one for the validation data. A data generator is capable of loading the required

amount of data (a minibatch of images) directly from the source folder, converting them into training data (fed to the model) and training targets (a vector of attributes—the supervision signal). Our training data are augmented via a number of random transformations in order to prevent duplicate images, that is, the model will never see the exact same picture twice. This has been done to prevent overfitting and improve the generalization of the model. The following table represents the settings of the deployed image augmentation (Table 3).

Table 3. Settings for Chest X-ray Image Augmentation.

Method	Setting
Rotation range	40
Width shift	0.2
Rescale	1.0/255
Height shift	0.2
Zoom range	0.3
Horizontal flip	True

The rotation range represents the range of the random rotations of the images during training, i.e., 40 degrees. The width shift represents the horizontal image translation by 0.2%, and the height shift represents the vertical image translation by 0.2%. The ratio of the zoom range randomly zooming in on images is 0.2%. Last, the images were flipped horizontally. We have done this using "keras.preprocessing.image.ImageDataGenerator" class. This class allows for configuring random transformations and normalization operations on image data during training. In addition, instantiate generators of augmented image batches via.flow (data, labels) or.flow_from_directory (directory). Detailed documentation about this class has been provided in the Keras documentation [27]. We have used the fit_generator, evaluate_generator, and predict_generator generators with the Keras model that accepts the data generators as inputs. Our generators are able to load the required amount of data directly from the source directory. Regarding the three subsets of our data set (test, training, validation), the training directory consists of 5218 chest X-ray images, the test directory consists of 526 images, and the validation directory consists of 18 images.

For symptom analysis, we collected data on the symptoms from detailed research work on COVID-19 patients that was published in March 2020 [24]. This research [24] was based on a population infected with COVID-19 in Wuhan, China. The data sample was 53,000 patients who were infected by COVID-19. We also collected a sample of the symptom data of 500 Bangladeshi patients who tested positive for COVID-19. These data were collected manually from Bangladeshi top newspapers, by contacting patients over the phone, and from patients interviewed via social media. For example, patient 1's data (she was admitted to the Kuwait Bangladesh Friendship Government Hospital for more than 1 week) were collected over the phone by talking with the patient, and patient 3's data were collected from a Bangladeshi top newspaper [28]. Out of these 500 patients, there were 4 patients who did not have any COVID-19 symptoms. Table 4 shows five patient data points as an overview.

Table 4. Infected patient symptom dataset from Bangladeshi people.

Patients	Symptoms	COVID-19 Status	State, Country	Gender
Patient 1	Fever, shortness of breath, cough, sore throat, muscle pain, and hyposmia (having difficulties smelling food and other things)	Positive	Dhaka, Bangladesh	Female
Patient 2	Fever, sore throat, shortness of breath, body chills, cough, muscle pain, and cannot eat food	Positive	Dhaka, Bangladesh	Female
Patient 3	Fever, shortness of breath, cough, muscle pain, and kidney problems	Positive	New York, NY, USA (Bangladeshi Immigrant)	Male
Patient 4	Fever, cough, sore throat, and muscle pain	Positive	New York, NY, USA (Bangladeshi Immigrant)	Male
Patient 5	Muscle pain, fatigue (tiredness), fever, and cough	Positive	Gaibandha, Bangladesh	Male

2.4.2. Applied Proposed CNN Model

The CNN algorithm resulted in an optimal solution by classifying abnormal (pneumonia labeled) and normal frontal chest X-ray images. Figure 6 provides an overview of the training, validation, and testing of the chest X-ray images.
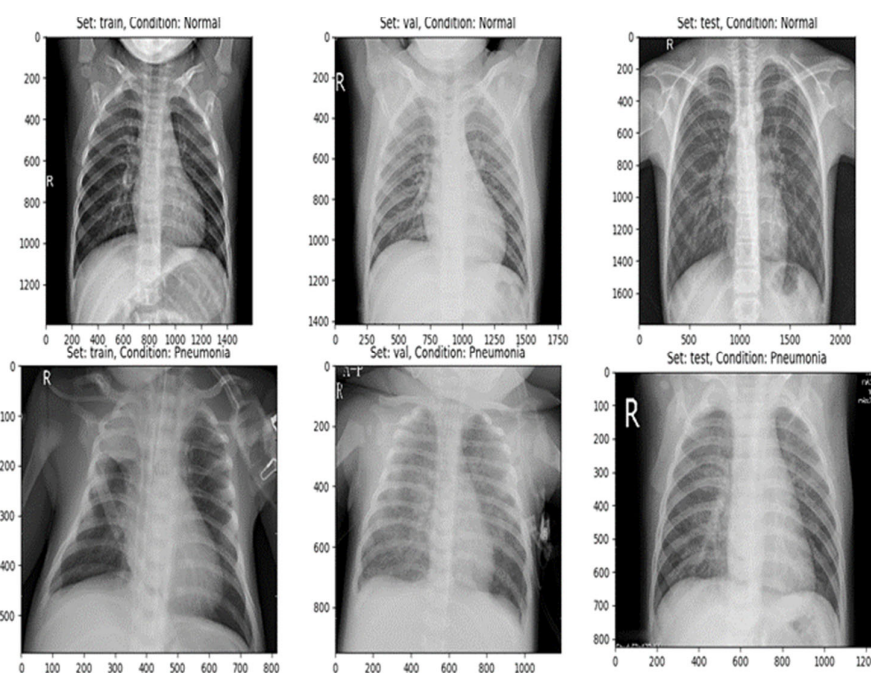
Figure 6. An overview of the training, validation, and testing of frontal chest X-ray images.

We conducted the experiment by following the CNN model architecture that is described in the methodology section (Figure 2). The classifier of the model is placed at the end of the proposed convolutional neural network (CNN) model. It is essentially an artificial neural network (ANN) regularly called a dense layer. To perform computations, like any other classifier, this classifier requires individual features (vectors). Therefore, for the classifiers, the feature extractor (CNN part) output is converted into a 1D feature vector. This procedure is known as flattening. In flattening, the output of the convolution activity is flattened to create one long feature vector for the dense layer to use in its last

classification process. The classification layer contains three dense layers of size 512, 128, and 64, respectively, three dropout rates of 0.7, 0.5, and 0.3, a ReLU (rectified linear unit) between the three dense layers, and a sigmoid activation function that performs the classification tasks and a flattened layer. We have used the "binary_crossentropy" loss to train our model, and we have used the "adam" optimizer. Appendix A explains some of the commonly used optimizers.

The experiment was conducted many times to check, evaluate and validate the effectiveness of the proposed procedure. The parameters and hyperparameters were deliberately tuned to increase the performance of the model. This study reports only the most valid result where we have obtained different results.

2.4.3. Applied Proposed SymptomNet Algorithm

The experiment is constructed into two parts: the 1st part is constructed with the default "symptom weight table" (Table 1), and the 2nd part is constructed with a specific country's (Bangladesh) "symptom weight table". The threshold value of the equation is changeable because it depends on the symptoms. Since COVID-19 can mutate, it is able to change its symptomatic characteristics. The following table is derived for calculating the threshold based on our default weight Table 1.

The following Figure 7 represents the default threshold point based on Table 5.

Table 5. Default threshold calculation table.

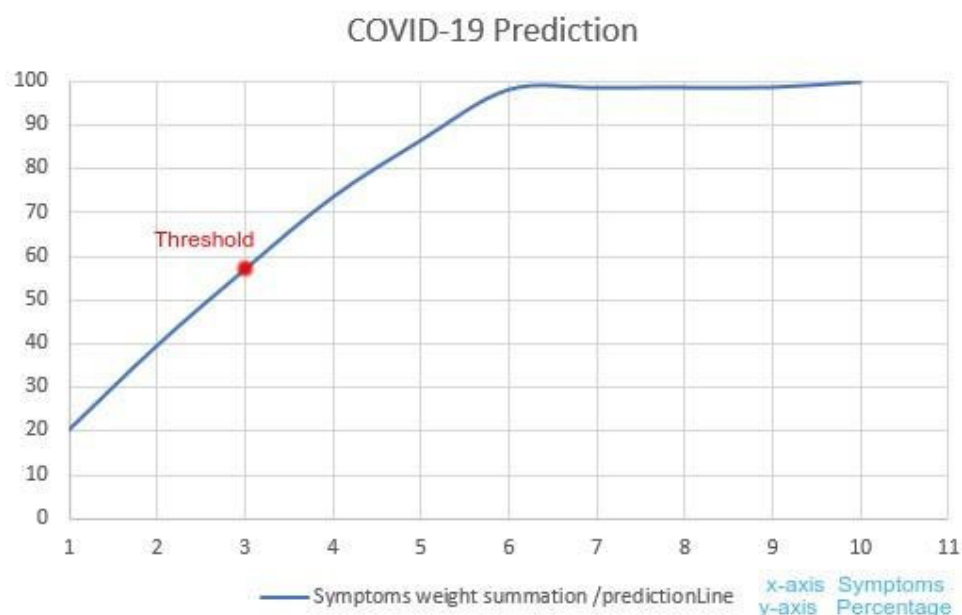| Clinical Symptoms	Individual Weight $Iw = (w \times 100)/Total\ Weight$	Weight Summation/Prediction Line $ws = Iw + ws_{previous}$
Fever	24.2	24.32
Cough	19.45	43.76
Fatigue	16.49	60.25
Shortness of breath	12.1	72.34
Muscle pain	7.11	79.45
Chill	7.11	86.56
Dizziness	4.4	90.96
Headache	3.09	94.05
Sore throat	2.13	96.18
Nausea or vomiting	1.61	97.79
Diarrhea	1.55	99.34
Nasal congestion	0.76	100

Figure 7. Graphical view of the default threshold point.

When Equation (1) crosses the threshold point, we can conclude that the patient may have a COVID-19 infection. As the result increases after the threshold, the probability of a possible COVID-19 infection will increase.

In this part, we have applied our collected Bangladeshi COVID-19-positive patient symptom dataset (see Table 4 in the data preprocessing section) to our derived Equations (1) and (2). Based on this dataset, fever was a symptom 92% of the time, the cough was a symptom 86% of the time, body chills were a symptom 78% of the time, shortness of breath was a symptom 74% of the time, muscle pain was a symptom 58% of the time, a sore throat was a symptom 52% of the time, and the other four symptoms were symptoms 2% of the time. First, we generated the symptom weight table from this dataset. The symptom weight Table 6 is shown below.

Table 6. Symptom weight table of Bangladeshi patients.

Clinical Symptoms	Symptom Weights
Fever	92
Cough	86
body chills	78
Shortness of breath	74
Muscle pain	58
Sore throat	52
Fatigue	2
Hyposmia (having difficulties smelling food and other things)	2
Cannot eat food	2
Kidney problem	2
Total Weight	448

Table 7 is derived to calculate the threshold point based on the symptom weight Table 6 of Bangladeshi patients.

Table 7. Threshold calculation table for the Bangladeshi patient symptom weight dataset.

Clinical Symptoms	Individual Weight $Iw = (w \times 100)/Total\ Weight$	Weight Summation/Prediction Line $ws = Iw + ws_{previous\_}$
Fever	20.54	20.54
Cough	19.19	39.74
body chills	17.41	**57.15 (Threshold point)**
Shortness of breath	16.52	73.66
Muscle pain	12.95	86.62
Sore throat	11.61	98.23
Fatigue	0.045	98.67
Hyposmia (having difficulties smelling food and other things)	0.045	98.72
Cannot eat food	0.045	98.77
Kidney problem	0.045	100

The following Figure 8 represents the threshold point for Bangladeshi patients based on Table 7.

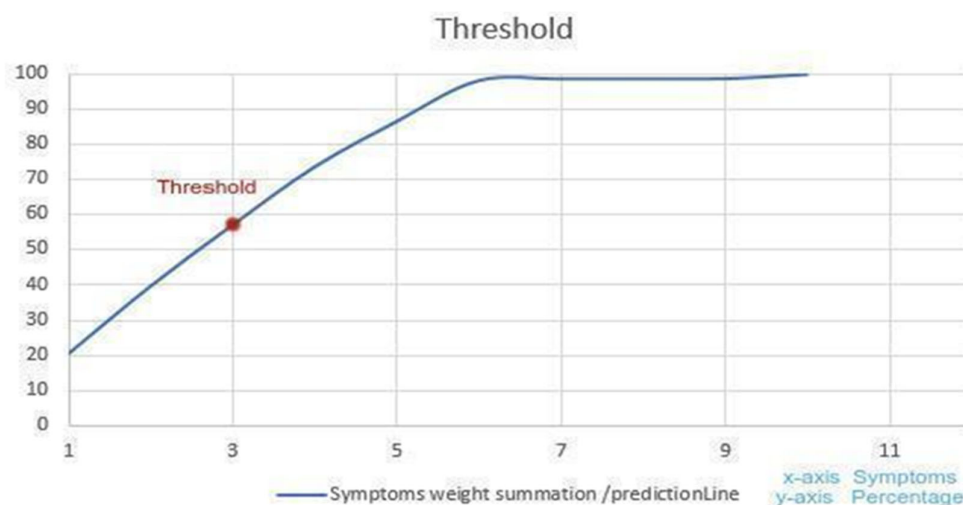Detailed results of the experiments are shown and described in the "Results" section.

Figure 8. Threshold point for the Bangladeshi patient symptom weight dataset.

3. Results

This proposed model is a combination of CNN-based frontal chest X-ray image classification and our own developed algorithm (SymptomNet). Our results are discussed in three parts. In the first two parts, we have provided the results of the individual algorithms (the CNN model and the SymptomsNet algorithm, respectively) in detail to provide a better understanding. In the last part, we have provided our full model results.

As clarified above, methods, such as data augmentation, varying the learning rate, and annealing, were used to help fit the small dataset into a deep convolutional neural network architecture. These methods were performed in order to acquire generous outcomes, as shown in Figures 9 and 10. The final results obtained are training loss = 0.1311, training accuracy = 0.9596, validation loss: 0.2754, and validation accuracy of 0.9273. The overall scenario has been shown in Figure 11.
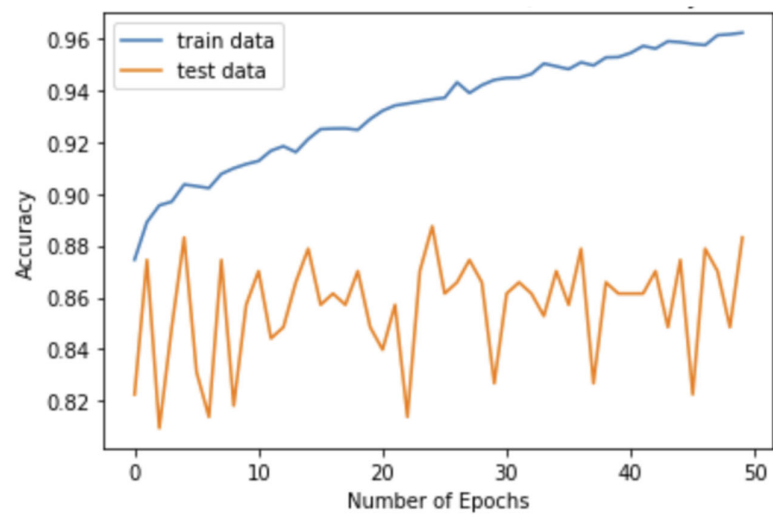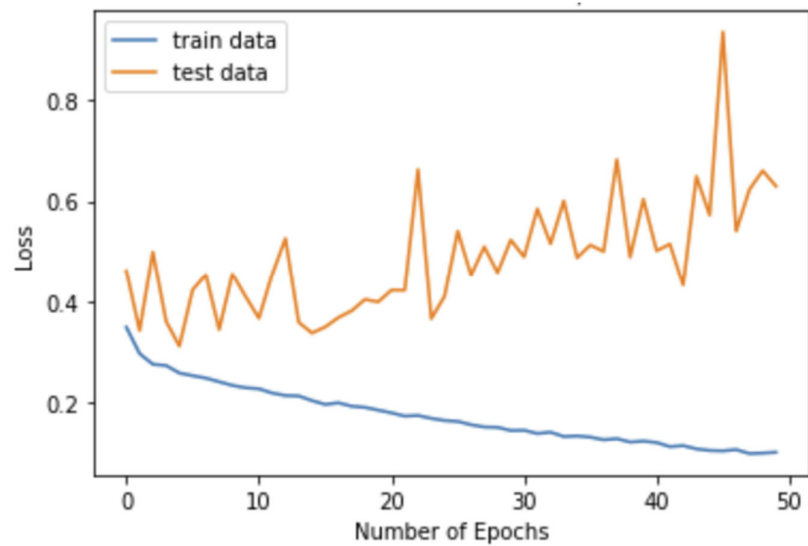
Figure 9. Model Accuracy vs. Epoch.

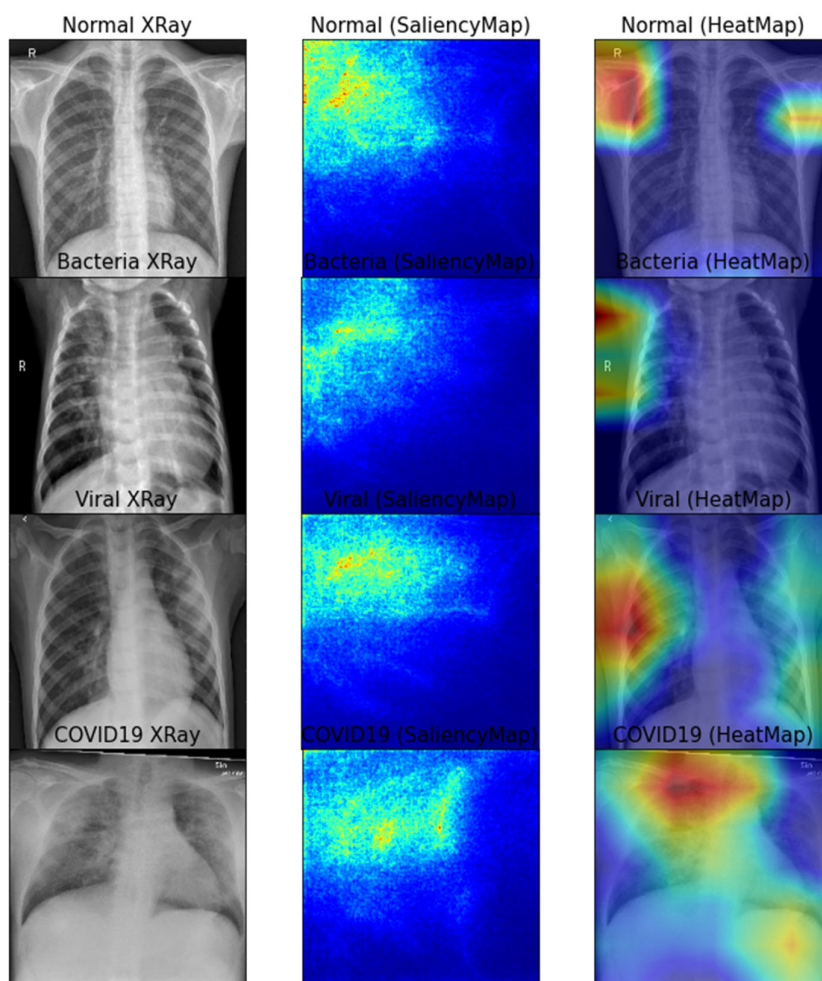
Figure 10. Model Loss vs. Epoch.

Figure 11. Visualization of general XRay vs Saliency Map Vs HeatMap. From top to bottom represents respectively - Normal XRay, Bacteria XRay, Viray XRay, and COVID19 XRay.

Figure 12 shows the confusion matrix and related parameters used to evaluate the model.
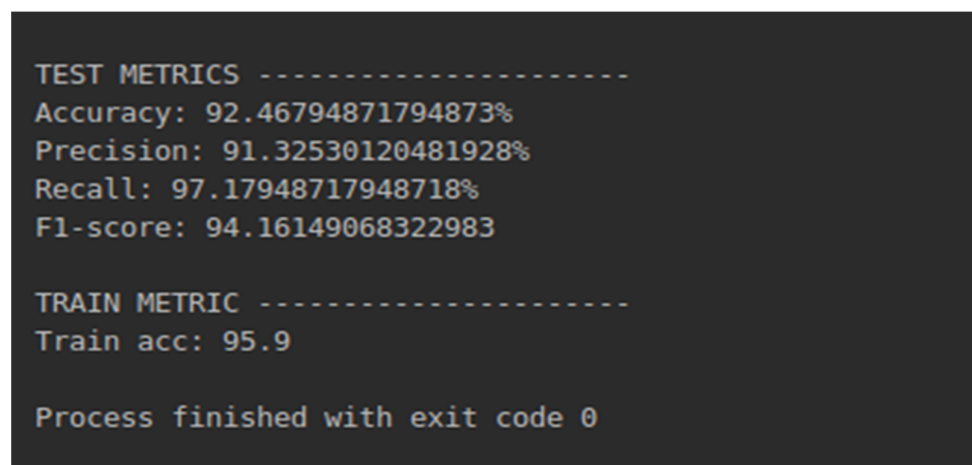


```
TEST METRICS ---------------------
Accuracy: 92.46794871794873%
Precision: 91.32530120481928%
Recall: 97.17948717948718%
F1-score: 94.16149068322983

TRAIN METRIC ---------------------
Train acc: 95.9

Process finished with exit code 0
```

Figure 12. Confusion Matrix.

This model's training accuracy is 95.9%, and it achieves a test accuracy of 92.467%, which indicates that it is a good model. With this model, we were able to determine lung infections (pneumonia), which is one of the major situations for COVID-19 patients.

In this part, we will show the results of our SymptomNet algorithm. We generated all the prediction results of the 500 Bangladeshi COVID-19-positive patients using Equation (1). Then, we compared the predicted results to the threshold point. The threshold point was generated using Equation (2). From the experiment, we found that this method achieved 97% (485 successful predictions out of 500) prediction accuracy. In Table 8, we have displayed the first 11 results for the 500 Bangladeshi COVID-19-positive patients.

Figures 13 and 14 are representing the correlations and p-values respectively. The role of "p" values and "r" values is to find out the best correlation and rank the feature to select the best features of the datasets [29]. The analysis of the correlation between the symptoms and COVID-19 status revealed that 10 out of 13 symptoms were significantly correlated to the COVID-19 disease. Specially, fever, cough, body chills, shortness of breath, muscle pain, sore throat were shown to be significantly related to the COVID-19 disease ($r = 0.20$; $p = 0.001$, $r = 0.20$; $p < 0.001$, $r = 0.22$; $p < 0.001$, $r = 0.16$; $p < 0.001$, $r = -0.45$; $p < 0.001$, $r = -0.35$; $p < 0.001$, respectively) from the developed symptoms in a patient. Also, the p-value curve showed strong correlations ($p < 0.05$) of the symptoms. p-values of other symptoms, such as fatigue, hyposmia, anorexia, kidney problem, dizziness, nausea, or vomiting are 0.319, 0.014, 0.318, 0.083, 0.001, <0.001, respectively.
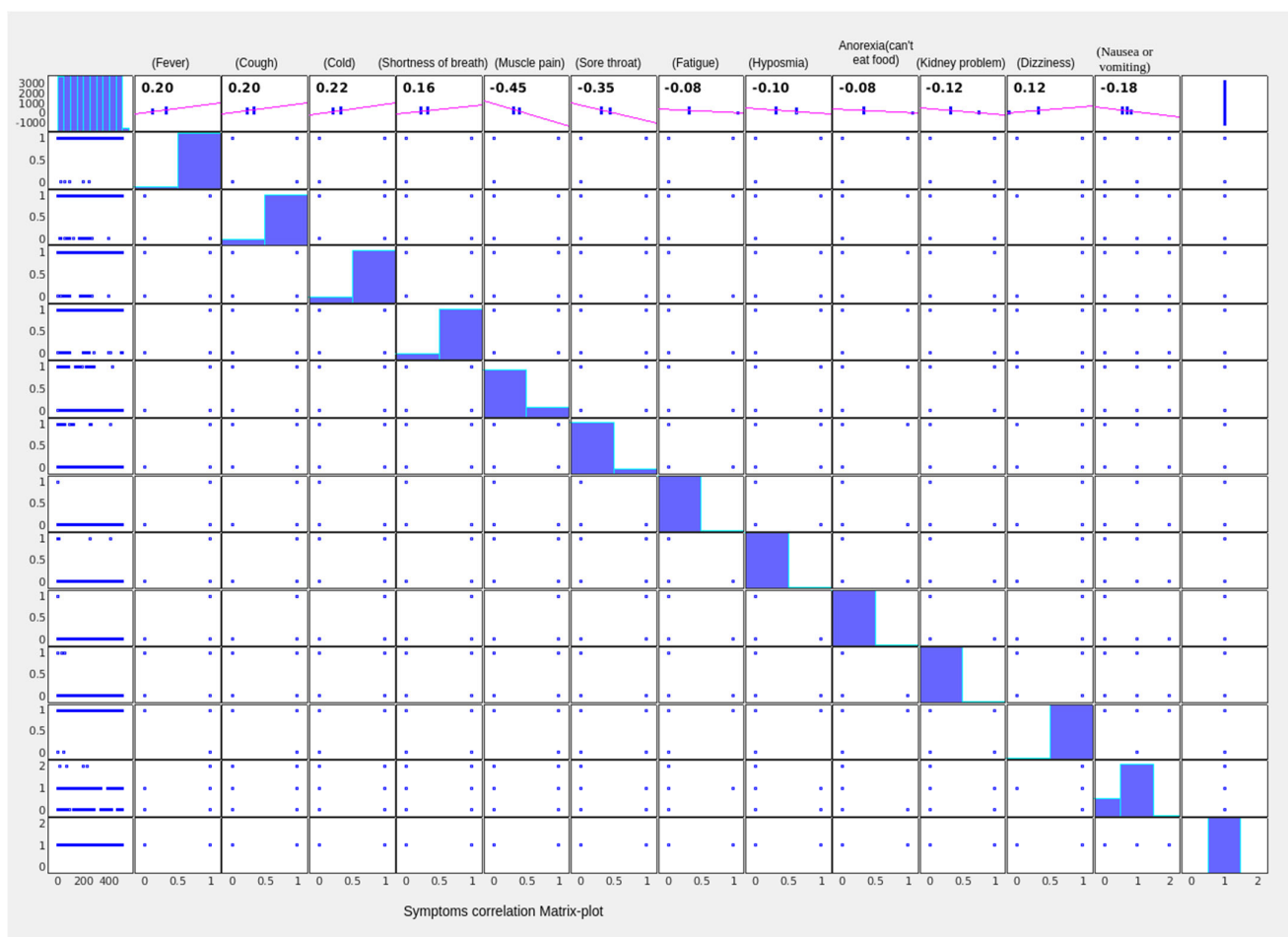

Figure 13. Correlation matrix-plot for symptoms. The coefficient of correlation of each feature and COVID-19 status is shown at the top of the plot.
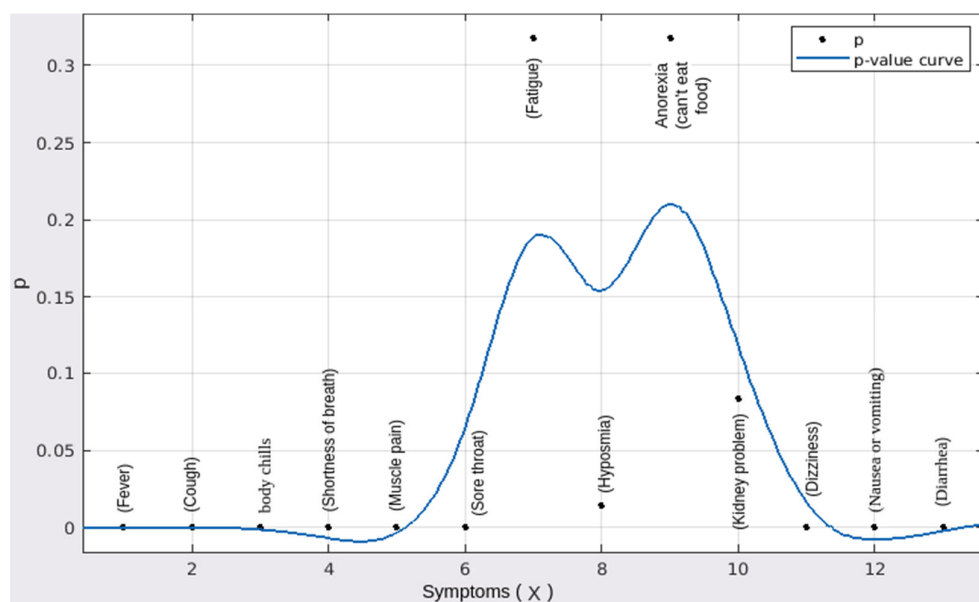
Figure 14. Graphical representation of *p*-values.

Table 8. Bangladeshi patients' prediction results vs. original results.

Patient	Threshold	Our Prediction (C_{pr})	Original COVID-19 Status	Prediction Accuracy
patient 1	57.15	80.055 (positive)	positive	Correct
patient 2	57.15	98.265 (positive)	positive	Correct
patient 3	57.15	69.245 (positive)	positive	Correct
patient 4	57.15	64.29 (positive)	positive	Correct
patient 5	57.15	57.725 (positive)	positive	Correct
patient 6	57.15	70.22 (positive)	positive	Correct
patient 7	57.15	86.61 (positive)	positive	Correct
patient 8	57.15	86.61 (positive)	positive	Correct
patient 9	57.15	86.61 (positive)	positive	Correct
patient 10	57.15	98.265 (positive)	positive	Correct
patient 11	57.15	54.57 (negative)	positive	Wrong

Figure 15 shows the graphical view of our results, where we have displayed the threshold point and the prediction results of individual patients. Patient 2 and patient 10 had the highest predicted percentages (98.28%), and patient 11 had the lowest predicted percentage (54.25%), which was below the threshold point.
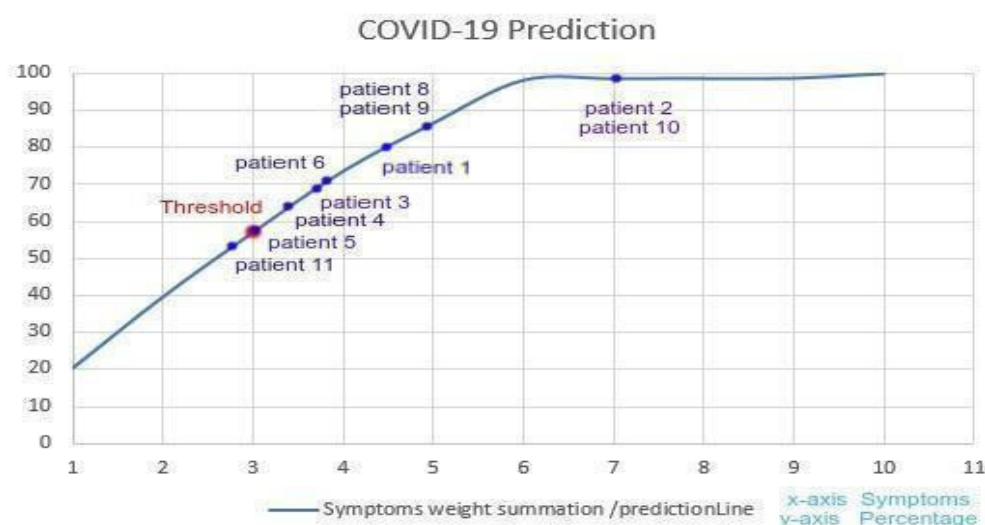
Figure 15. COVID-19 prediction (*C$_{pr}$*).

According to the Canadian PHAC [30], the top symptoms of COVID-19 in Canada are fever, cough, difficulty breathing, and pneumonia in both lungs. According to the UK's NHS [31], the top symptoms of COVID-19 in the UK are cough, high temperature, and shortness of breath. According to the Australian Government Department of Health [32], the top symptoms of COVID-19 in Australia are fever, coughing, shortness of breath, sore throat, and fatigue. According to the CDC in the USA [33], they are facing similar symptoms for COVID-19. Based on the mentioned countries' health organizations, our equation is fully able to identify the possible COVID-19 patients according to their symptoms.

In this last part of the Results section, we have constructed the final results of our full model. We ran the final test on 30 COVID-19-positive patients. To test and generate the results, we used patients' frontal chest X-ray images and symptoms. These data were collected from Bangladesh. Since we are using Bangladeshi patients' data, for the "SymptomNet" algorithm, we are not using the default "symptom weight table". We are using the "symptom weight table", which was made based on Bangladeshi patients' data (see Table 9 in the Results section). Since the previous two parts of this results section have described the detailed procedure, in this section, we have shown the final output only. Table 9 shows the results.

Table 9. Experimental results of the developed model.

Patient No	CNN Classified as	CNN Accuracy	SymptomNet Threshold	SymptomNet PREDICTION (%)	SymptomNet Prediction Accuracy	Our Model's Prediction	Patient's Original COVID-19 Status	Model Prediction Accuracy Status
1	pneumonia			98.21 (positive)		High possibility	Positive	**Correct**
2	pneumonia			74.1 (positive)		High possibility	Positive	**Correct**
3	pneumonia			98.21 (positive)		High possibility	Positive	**Correct**
4	pneumonia			98.21 (positive)		High possibility	Positive	**Correct**
5	normal	95.9%	57.15	57.59 (positive)	97%	Low possibility	Positive	**Correct**

6	pneumonia	85.26 (positive)	High possibility	Positive	**Correct**
7	normal	36.6 (negative)	No Infection	Positive	**Wrong**
8	pneumonia	98.21 (positive)	High possibility	Positive	**Correct**
9	normal	57.15 (positive)	Low possibility	Positive	**Correct**
10	normal	74.1 (positive)	High possibility	Positive	**Correct**
11	pneumonia	83.56 (positive)	High possibility	Positive	**Correct**
12	normal	33.26 (negative)	No Infection	Negative	**Correct**
13	pneumonia	96.32 (positive)	High possibility	Positive	**Correct**
14	pneumonia	72.31 (positive)	High possibility	Positive	**Correct**
15	pneumonia	95.71 (positive)	High possibility	Positive	**Correct**
16	pneumonia	93.32 (positive)	High possibility	Positive	**Correct**
17	normal	36.6 (negative)	No Infection	Positive	**Wrong**
18	pneumonia	96.27 (positive)	High possibility	Positive	**Correct**
19	normal	59.75 (positive)	Low possibility	Positive	**Correct**
20	normal	84.29 (positive)	High possibility	Positive	**Correct**
21	pneumonia	91.36 (positive)	High possibility	Positive	**Correct**
22	normal	35.36 (negative)	No Infection	Negative	**Correct**
23	pneumonia	66.32 (positive)	Low possibility	Negative	**Wrong**
24	pneumonia	82.71 (positive)	High possibility	Positive	**Correct**
25	pneumonia	97.51 (positive)	High possibility	Positive	**Correct**
26	pneumonia	96.62 (positive)	High possibility	Positive	**Correct**
27	normal	66.6 (positive)	Low possibility	Positive	**Correct**
28	pneumonia	86.37 (positive)	High possibility	Positive	**Correct**
29	normal	58.85 (positive)	Low possibility	Positive	**Correct**
30	normal	84.29 (positive)	High possibility	Positive	**Correct**

4. Discussion

Figures 9 and 10 show that the obtained training loss = 0.1311, training accuracy = 0.9596, validation loss = 0.2754, and validation accuracy of 0.9273 while classifying X-ray images as infected (pneumonia) and normal. Our CNN algorithm achieves a training accuracy of 95.9% (Figure 12). The results indicate that our CNN model serves its purpose for image classification. The novel part of this research article is the SymptomNet algorithm. This algorithm is designed to predict COVID-19 patients by analyzing patients' symptoms. The SymptomNet algorithm achieves 97% accuracy at predicting COVID-19 infections. The combined model (CNN image classification and SymptomNet) provided more accurate results for COVID-19 identification. Few research articles [34–36] that have been published recently have focused on identifying pneumonia from X-ray images, which can indicate COVID-19 infections. Diagnosing pneumonia cannot be the sole determinant of COVID-19 identification. One research article [37] showed that 138 million children were infected with pneumonia from 2000–2015. This indicates that, per year, on average, 9.2 million children are infected by pneumonia. Therefore, COVID-19 symptom analysis is necessary along with pneumonia identification. As a result, we have developed the novel SymptomNet algorithm to provide nearly perfect results. According to our final result (Table 9), we were able to correctly identify 27 patients out of 30 with COVID-19 infections. We will attempt to collect more data to extend this research work. This model overall achieves excellent performance and is capable of identifying possible COVID-19 patients. Because of its effectiveness and short execution time at identifying possible COVID-19 patients, we believe that this model can contribute to improving the crisis of COVID-19 situations worldwide. The research work was limited by the depth of the data. Future work could further improve the determination of the threshold point. Our future work also relied on optimized model-based neural network compression [38].

5. Conclusions

Although the model in the current study achieves high accuracy at predicting COVID-19-infected patients, there are a few limitations to consider when interpreting the results. The amount of data we use for this work was not enough to fully prove this model. We used 5300 frontal chest X-ray images for training and then classifying lung infections (COVID-19 pneumonia), but the model could be improved if the number of X-ray images increased. The X-ray images did not come directly from COVID-19-related patients, which is a drawback for this model. In the analysis of the symptoms, we have used the data of 500 patients who are COVID-19-positive to evaluate our algorithm and equations, and we have achieved 97% accuracy; however, this accuracy may be increased to near 100% if the amount of data is larger. Our main aim was to detect COVID-19 patients by our own developed mathematical model. Furthermore, the calculation of the threshold point can be further optimized with more research work. More data would be good enough to train our model to get better accuracy than we have now. Another limitation of the study was that we were not able to compare to other similar illnesses.

In summary, we have figured out a model to detect COVID-19 patients based on chest X-ray images through CNN along with symptom analysis through a mathematical model analysis.

Author Contributions: M.H.U.: conceptualization, methodology, software, validation, formal analysis; M.N.H.: resources, data curation, data analysis, methodology, writing—original draft & review; M.S.I.: visualization, formal analysis; M.A.A.Z.: visualization, formal analysis; S.-H.Y.: supervision. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: This study was approved by the Institutional Review Board of Enam Medical College Hospital, Dhaka, Bangladesh [Ref No: EMC.BD/+88-02-7743662]. All the

participants were informed about the purpose, nature, and procedure of the study, and they were also informed that they have full rights to withdraw their data at any time.

Appendix A

Table A1 shows different kinds of optimization algorithms and their working process.

Table A1. Different kinds of optimization algorithms and their working process.

Optimizer Algorithm	Optimization Process
First-Order Optimization	These algorithms minimize or maximize a loss function E(x) by using its gradient values. Gradient descent is the most widely used first-order optimization algorithm. From the first-order derivative, it can determine whether the function is increasing or decreasing at a particular point. It gives us a tangential line to a point on its error surface.
Second-Order Optimization	The second-order derivative, also known as the Hessian, minimizes or maximizes the Loss function. It uses the second-order partial derivatives matrix. The second order is not used much because it is costly to compute. The second-order derivative represents the function's curvature by determining whether the first derivative is increasing or decreasing. The second-order derivative provides a quadratic surface. This quadratic surface touches the curvature of the error surface.
Stochastic gradient descent	Stochastic gradient descent conducts parameter updating for each training example. This technique is usually faster. Stochastic gradient descent performs by updating the parameters one at a time. $$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^{(i)}; y^{(i)}),$$ where $\{x^{(i)}; y^{(i)}\}$ are the training examples.
Adagrad Optimizer	Approach of the AdaGrad optimizer is to use a different learning rate for each and every parameter θ at a time step based on the previous gradients that were calculated for that parameter. "It modifies the approach of general learning rate η at each time step *t* for every parameter θ

	based on the previous gradients that *i* have been computed for θ*i*" [39]. $$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii+\epsilon}}} \cdot g_{t,i}.$$
AdaDelta	AdaDelta tends to eliminate the decaying learning rate problem of AdaGrad. Basically, it is an extension of AdaGrad. "Adadelta limits the window of accumulated past gradients to some fixed size *w*, Instead of accumulating all previous squared gradients" [39]. $$E[g^2]_t = \gamma.E[g^2]_{t-1} + (1-\gamma).g^2{}_t,$$ set γ to a similar value as the momentum term, around 0.9. $$\Delta\theta_t = -\eta \cdot g_{t,i}.$$ $$\theta_{t+1} = \theta_t + \Delta\theta_t.$$ $$\Delta\theta_t = -\frac{\eta}{\sqrt{E[g^2]_t+\epsilon}} g_t \Rightarrow \Delta\theta_t = -\frac{\eta}{RMS[g_t]} g_t$$
Adam	Adaptive Moment Estimation (Adam) is another technique that computes adaptive learning rates for each parameter. Like AdaDelta technique, in addition to storing the exponentially decay normal of previous squared Gradients, Adam likewise keeps an exponentially decaying normal of previous gradients M_t. $$\widehat{m_t} = \frac{m_t}{1-\beta_1^t}$$ $$\widehat{v_t} = \frac{v_t}{1-\beta_2^t}$$ These are the formulas for the first moment (mean) and the second moment (the variance) of the gradients. The final formula for the parameter update is - $$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\widehat{v_t}}+\epsilon} + \widehat{m_t}$$

References

1. Bazell, R. How Genetic Mutations Turned the Coronavirus Deadly. Available online: http://nautil.us/issue/83/intelligence/how-genetic-mutations-turned-thecoronavirus-deadly?fbclid=IwAR3oUg2cDDqCGz4SmZVduxtCxUaPeBejnyUPkJtg34wrQTFec−OBzBz2×4 (accessed on accessed on 28 April 2020).
2. World Health Organization (WHO). WHO Director-General's Opening Remarks at the Media Briefing on COVID-19. Available online: https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-COVID-19---16-march-2020 (accessed on 16 March 2020).
3. Sudre, C.H.; Lee, K.; Ni Lochlainn, M.; Varsavsky, T.; Murray, B.; Graham, M.S.; Menni, C.; Modat, M.; Bowyer, R.C.E.; Nguyen, L.H.; et al. Symptom clusters in Covid19: A potential clinical prediction tool from the COVID Symptom study app. *medRxiv* **2020**. https://doi.org/10.1101/2020.06.12.20129056.
4. Dong, E.; Du, H.; Gardener, L. An interactive web-based dashboard to track COVID-19 in real-time. *Lancet Infect. Dis.* **2020**, *20*, 533–534.
5. World Health Organization. Coronavirus Disease 2019 (COVID-19). Situation Report-72. 1 April 2020. Available online: https://apps.who.int/iris/bitstream/handle/10665/331685/nCoVsitrep01Apr2020-eng.pdf (accessed on 16 March 2020).
6. Xu, Z.; Shi, L.; Wang, Y.; Zhang, J.; Huang, L.; Zhang, C.; Liu, S.; Zhao, P.; Liu, H.; Zhu, L. Pathological findings of COVID-19 associated with acute respiratory distress syndrome. *Lancet Respir. Med.* **2020**, *8*, 420–422. https://doi.org/10.1016/S2213-2600(20)30076-X.
7. Wu, Z.; McGoogan, J.M. Characteristics of and important lessons from the coronavirus disease 2019 (COVID-19) outbreak in china: Summary of a report of 72 314 cases from the Chinese center for disease control and prevention. *JAMA* **2020**, *323*, 1239–1242. https://doi.org/10.1001/jama.2020.2648.

8. Chen, H.; Guo, J.; Wang, C.; Luo, F.; Yu, X.; Zhang, W.; Li, J.; Zhao, D.; Xu, D.; Gong, Q. Clinical characteristics and intrauterine vertical transmission potential of COVID-19 infection in nine pregnant women: A retrospective review of medical records. *Lancet* **2020**, *395*, 420–422. https://doi.org/10.1016/S0140-6736(20)30360-3.

9. Rao, A.S.S.; Vazquez, J.A. Infection Control & Hospital Epidemiology. Identification of COVID-19 Can Be Quicker through Artificial Intelligence Framework Using a Mobile Phone-Based Survey in the Populations When Cities/Towns Are under Quarantine; Cambridge University Press: Cambridge, UK, 2020; pp. 1–18. https://doi.org/10.1017/ice.2020.61.

10. Fang, Y.; Zhang, H.; Xie, J.; Lin, M.; Ying, L.; Pang, P.; Ji, W. Sensitivity of chest ct for COVID-19: Comparison to rt-pcr. *Radiology* **2020**, *296*, 200432. https://doi.org/10.1148/radiol.2020200432.

11. Liu, Y.; Gayle, A.A.; Wilder-Smith, A.; Rocklöv, J. The reproductive number of COVID-19 is higher compared to sars coronavirus. *J. Travel Med.* **2020**, *27*, taaa021. https://doi.org/10.1093/jtm/taaa021.

12. Pandey, R.; Gautam, V.; Bhagat, K.; Sethi, T. A machine learning application for raising wash awareness in the times of COVID-19 pandemic. *arXiv* **2020**, arXiv:2003.07074.

13. World Health Organization (WHO). Available online: https://www.who.int/ (accessed on 28 April 2020).

14. Pan, F.; Ye, T.; Sun, P.; Gui, S.; Liang, B.; Li, L.; Zheng, D.; Wang, J.; Hesketh, R.L.; Yang, L.; et al. Time course of lung changes on chest ct during recovery from 2019 novel coronavirus (COVID-19) pneumonia. *Radiology* **2020**, 200370. https://doi.org/10.1148/radiol.2020200370].

15. Aledhari, M.; Joji, S.; Hefeida, M.; Saeed, F. Optimized CNN-based Diagnosis System to Detect the Pneumonia from Chest Radiographs. In Proceedings of the 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), San Diego, CA, USA, 18–21 November 2019; pp. 2405–2412. https://doi.org/10.1109/BIBM47256.2019.8983114.

16. Chouhan, V.; Singh, S.K.; Khamparia, A.; Gupta, D.; Tiwari, P.; Moreira, C.; Damaševičius, R.; De Albuquerque, V.H.C. A novel transfer learning-based approach for pneumonia detection in chest X-ray images. *Appl. Sci.* **2020**, *10*, 559. https://doi.org/10.3390/app10020559.

17. Toğaçar, M.; Ergen, B.; Cömert, Z.; Özyurt, F. A deep feature learning model for pneumonia detection applying a combination of mRMR feature selection and machine learning models. *IRBM* **2020**, *41*, 212–222. https://doi.org/10.1016/j.irbm.2019.10.006.

18. Rajpurkar, P.; Irvin, J.; Zhu, K.; Yang, B.; Mehta, H.; Duan, T.; Ding, D.; Bagul, A.; Langlotz, C.; Shpanskaya, K. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *arXiv* **2017**, arXiv:1711.05225.

19. Pankratz, D.G.; Choi, Y.; Imtiaz, U.; Fedorowicz, G.M.; Anderson, J.D.; Colby, T.V.; Myers, J.L.; Lynch, D.A.; Brown, K.K.; Flaherty, K.R. Usual interstitial pneumonia can be detected in transbronchial biopsies using machine learning. *Ann. Am. Thorac. Soc.* **2017**, *14*, 1646–1654. https://doi.org/10.1513/AnnalsATS.201612-947OC.

20. Lakhani, P.; Sundaram, B. Deep learning at chest radiography: Automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology* **2017**, *284*, 574–582. https://doi.org/10.1148/radiol.2017162326.

21. Stephen, O.; Sain, M.; Maduh, U.J.; Jeong, D.-U. An efficient deep learning approach to pneumonia classification in healthcare. *J. Healthc. Eng.* **2019**, *2019*, 4180949. https://doi.org/10.1155/2019/4180949.

22. Shin, H.-C.; Lu, L.; Kim, L.; Seff, A.; Yao, J.; Summers, R.M. Interleaved Text/image deep mining on a very large-scale radiology database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1090–1099. https://doi.org/10.1109/CVPR.2015.7298712 .

23. Wang, X.; Peng, Y.; Lu, L.; Lu, Z.; Bagheri, M.; Summers, R.M. Chestxray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2097–2106. https://doi.org/10.1109/CVPR.2017.369.

24. Zhao, X.; Zhang, B.; Li, P.; Ma, C.; Gu, J.; Hou, P.; Guo, Z.; Wu, H.; Bai, Y. Incidence, clinical characteristics, and prognostic factor of patients with COVID-19: A systematic review and meta- analysis. *medRxiv* **2020**. https://doi.org/10.1101/2020.03.17.20037572.

25. Chollet, F. Keras. 2015. Available online: https://github.com/fchollet/keras (accessed on 16 March 2020).

26. Mooney, P. Chest X-ray Images (Pneumonia). Available online: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia (accessed on 28 April 2020).

27. Keras, Image Preprocessing. Available online: https://keras.io/preprocessing/image/ (accessed on 28 April 2020).

28. Chowdhury, I.; Alo, P. Available online: https://www.prothomalo.com/northamerica/article/1653129/ (accessed on 28 April 2020).

29. Hossain, M.N.; Uddin, M.H.; Thapa, K.; Zubaer, M.A.M.; Islam, M.S.;Lee, J.; Park, J.; Yang, S.-H. Detecting Cognitive Impairment Status Using Keystroke Patterns and Physical Activity Data among the Older Adults: A Machine Learning Approach. *J. Healthc. Eng.* **2021**, https://doi.org/10.1155/2021/1302989.

30. Government of Canada, Symptoms of COVID-19. Available online: https://www.canada.ca/en/publichealth/services/diseases/2019- novel-coronavirusinfection/symptoms.htmls (accessed on 28 April 2020).

31. Government of United Kingdom, Coronavirus (COVID-19). Available online: https://www.nhs.uk/conditions/coronaviruscovid-19/ (accessed on 28 April 2020).

32. Department of Health, Australian Government. Coronavirus (COVID-19) Health Alert. Available online: https://www.health.gov.au/news/health-alerts/novel-coronavirus-2019-ncov-health-alert (accessed on 28 April 2020).

33. Centers of Disease Control and Prevention, Symptoms of Coronavirus. Available online: https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html (accessed on 28 April 2020).

34. Ozturk, T.; Talo, M.; Yildirim, E.A.; Baloglu, U.B.; Yildirim, O.; Rajendra Acharya, U. Automated detection of COVID-19 cases using deep neural networks with X-ray images. *J. Comput. Biol. Med.* **2020**, *121*, 103792. https://doi.org/10.1016/j.compbiomed.2020.103792.

35. Asif, S.; Wenhui, Y.; Jin, H.; Tao, Y.; Jinhai, S. Classification of COVID-19 from Chest X-ray images using Deep Convolutional Neural Networks. *medRxiv* **2020**. https://doi.org/10.1101/2020.05.01.20088211.

36. Purohit, K.; Kesarwani, A.; Kisku, D.R.; Dalui, M. COVID-19 Detection on Chest X-Ray and CT Scan Images Using Multi-image Augmented Deep Learning Model. *BioRxiv* **2020**, 205567. https://doi.org/10.1101/2020.07.15.205567.

37. McAllister, D.A.; Liu, L.; Shi, T.; Chu, Y.; Reed, C.; Burrows, J.; Adeloye, D.; Rudan, I.; Black, R.E.; Campbell, H.; et al. Global, regional, and national estimates of pneumonia morbidity and mortality in children younger than 5 years between 2000 and 2015: A systematic analysis. *Lance Glob. Health* **2019**, *7*, e47–e57. https://doi.org/10.1016/S2214-109X(18)30408-X.

38. Uddin, M.H.; Ara, J.M.K.; Rahman M.H.; Yang, S.H. Neural network pruning: An effective way to reduce the initial network for deep learning based human activity recognition. In proceedings of the 2021 International Conference on Electronics, Communications and Information Technology (ICECIT), Khulna, Bangladesh; 14–16 September 2021, pp. 1–4. (doi: 10.1109/ICECIT54077.2021.9641226).

39. Ruder, S. An overview of gradient descent optimization algorithms. *arXiv* **2017**, arxiv:1609.04747.