SN Computer Science (2021) 2:304
https://doi.org/10.1007/s42979-021-00701-w

ORIGINAL RESEARCH

A Bi-fold Approach to Detect and Classify COVID-19 X-Ray Images and Symptom Auditor

Ahan Chatterjee[1] · **Swagatam Roy**[1] · **Sunanda Das**[2]

Received: 10 January 2021 / Accepted: 12 May 2021

Abstract
In this paper, we propose an ensemble-based transfer learning method to predict the X-ray image of a COVID-19 affected person. We have used a weighted Euclidean distance average as the parameter to ensemble the transfer learning model viz. ResNet50, VGG16, VGG19, Xception, and InceptionV3. Image augmentations have been carried out using generative adversarial network modelling. We took 784 training images, and 278 test images to validate our model accuracy, and the accuracy of our proposed model was around 98.67% for the training data set and 95.52% for the test data set. Along with that, we also propose a genetic algorithm optimized classification algorithm, to analyze the symptoms of COVID-19 for low, medium, and high-risk patients. The accuracy for the optimized set overshadowed the accuracy of un-optimized classification, and the optimized accuracy is as high as 88.96% for the optimized model. The novelty of this paper lies in the bi-sided model of the paper, i.e., we propose two major models, and one is the genetic algorithm optimized model to analyze the symptoms for a patient of varied risk and the other is to classify the X-ray image using an ensemble-based transfer learning model.

Keywords Genetic algorithm · Transfer learning · ResNet50 · COVID-19 · DCGAN · Naïve Bayes

Introduction

The arrival of COVID-19 shock the world with its arrival, first detected at Wuhan, Hubei province of China in the month of December. It was first noted as unknown pneumonia clustering in the region and later it spread rapidly and took the shape of a deadly pandemic. Initially, it has been assumed that the transmission is among the animals but soon it was proved that human transmission is also occurring, and from then social distancing became our new normal. The virus's nature is similar to previous attacks of SARS-CoV, and MERS through SARS-CoV2's spreading rate is higher than the former ones but the mortality rate is lower than those. The typical symptoms which are seen in the patients affected by the virus are fever, cough, fatigue, sore through, muscle pain, shortness of breath, etc.

Rapid testing has been suggested across all nations to cut the spread of the virus. The major aim is to separate the infected population part from the susceptible population part. The testing kit is the need of the hour to carry on the tests and one of the most common tests to detect the presence of nCoV is the reverse transcription–polymerase chain reaction (RT–PCR) test. Though the RT–PCR test is extremely costly in nature and major laboratories do not have access to these kits to carry on the test; thus, the speed of rapid testing is declining at a brisk pace. Whereas, on the other hand, the availability of X-ray is widely available across India, and thus, it can be a potential solution for rapid testing if it could be merged with our proposed model.

In this paper, we present an ensemble-based transfer learning model to easily classify the X-ray image into 2 major classes' COVID-19 positive or, COVID-19 negative. The patches start to appear from 0 to 2 days from the onset of the disease, and heavy patches are being created with the passing time. We have proposed a Euclidean average weights method to initialize the weights of the pre-trained models viz. ResNet50, VGG16, VGG19, Xception, and InceptionV3 while making the ensemble model classify the X-ray images with more accuracy. Initially, all the models are given the same importance and all the models are being assigned as

✉ Ahan Chatterjee
chatterjeeahan02@gmail.com

[1] Department of Computer Science and Engineering, The Neotia University, Sarisha, West Bengal, India

[2] Department of Computer Science and Engineering, SVCET, Chittoor, Andhra Pradesh, India

Published online: 28 May 2021

(1,1, 1,1, 1) in the vector space for the ensemble. Minimizing the false-negative count is one of the major aims while designing the classifier, and the ensembled proposed model has done that pretty well is can be shown using the confusion matrix. The pre-trained models are chosen that have a lower number of parameters to make our model computationally fast and economical keeping the accuracy factor in mind.

The images which have been passed through the classifier have been heavily pre-processed to provide the best image quality to maximize the prediction accuracy. At first, all the images were scaled to (224×224) size as those pre-trained models are trained in such dimensions. There was a lack of training data and the data set consisted of around 784 images; thus, we have data augmented the image using DCGAN architecture. The new synthetic images which have been created by the DCGAN architecture have been subsequently passed through as training images, thus diminish the class-imbalance problem for the classifier. The pre-processing followed by noise removal from the image using Gaussian filter model, then we have used shadow removal and Image Enhancement using Canny edge detection and histogram equalization. The lung portion is being segmented to analyze the match with higher sensitivity.

Along with that we have also presented a symptom analysis model. In this section the symptoms of the patients are taken into account viz. fever, tiredness, dry cough, sore throat, no symptoms, muscle pain, age, nasal congestion, running nose, and gender to classify them according to low, medium, and high severity. In this section, the classifier is being optimized using the Genetic Algorithm to get the fittest offspring from a series of a generation. The optimized model showed better results than the former one. We took 10 generations for the iteration and with that 20 offspring per generation has been taken into account for measuring the fitness factor. The proposed model is can be a groundbreaking result as we can easily predict the severity of the patients with the help of symptoms shown by the patients across the day.

The novelty of the paper lies in bi-fold modeling. On one hand, we have proposed a novel ensemble-based transfer learning method, where the proposed architecture surpasses the accuracy metrics with respect to the traditional models. In addition, on the other hand, we have proposed another classification model which can model the severity of a patient from the onset of the symptoms. Along with that, the classifier model has been optimized using the genetic algorithm to tune the accuracy of the model.

The paper is structured as "Literature Review" contains the literature review of the work, followed by ensemble-based proposed model in "Proposed Approach of Ensemble-Based Transfer Learning Model Using Euclidian Weighted Average" and Symptom classification analysis using genetic algorithm are proposed in "Symptom Analysis Using Hybrid Approach of Genetic Algorithm and Classifying Algorithm" and finally concluding remark and future scope of study in "Concluding Remark and Future Scope of Study"

Literature Review

A significant amount of work is being carried out in this field to help the frontline workers to some extent. Radiologists discovered major findings like Kong et al. [1] observed opaque inferior airspaces in the CT images of the lung. Yoon Kong et al. [2] observed opacity in the left lower lung region in nodal structure through CT images. Vascular dilation and several patches have been seen by Zhao et al. [3] with some irregular opacities in the lung region. Li and Xia [4] observed GGO and patches across the lungs of the COVID-19 affected patients, along with that they reported signs of air bronchogram. Along with that vascular expansion is also seen in CT images which became a common factor in the patients and most of them showing similar traits. Zhu et al. [5] reported that approximately 33% of chest CT scans have rounded patches. In a work, a convolutional neural network (CNN) model has been developed to classify the CT images in respective classes of COVID-19 positive or negative [6]. Another model has also been created to classify images and it worked with an accuracy of around 89% [7]. Recent works have also been done to classify images using CNN architectures. Hemdan et al. [8] proposed a CNN architecture consisting of 8 convolution layers. Wang and Wong [9] proposed another CNN model which yielded an accuracy of around 92.54% in classifying 3 classes of pneumonia viz. none, non-COVID patches, and COVID patches. Ioannis et al. [2] proposed a model with 224 images and they got an accuracy of around 93.48% in their test data set. Sethy and Behera [101] proposed a method to classify images using the SVM algorithm. Apart from these, there are several deep learning works, where they proposed the model using various transfer learning methods [11–17].

Feature engineering is an important aspect in the field of machine learning and artificial intelligence. Selecting the subset from the original set to increase the performance is the key factor and the genetic algorithm helps to select the best attributes with high accuracy [18]. In a genetic algorithm, each chromosome is considered a solution. The algorithm operates on population and selects the best possible genes out of it. John et al. [19] used the medical field in the medical field for optimizing chemotherapy and how an artificial immune system can be implemented is illustrated. There may be many features that may not be highly correlated to the target and so the feature selection is set not fixed and here feature selection using this algorithm plays an important role and get preference over sequential forward and backward selection. H. Kim et al. [20] used for the

non-linear optimization problem. It will highly improve the model [21, 22], and the removal of redundant features also helps to enhance the accuracy classification model. Hussein et al. in their work illustrated feature weighting and selection and used it in character recognition and pattern recognition [23]. Yang et al. [24] used it for feasibility testing of this algorithm in a neural network to improve the accuracy. Saidi et al. used big data and he stated that if the data is too large. For big data, it is very difficult to use this algorithm and thus parallel selection using this algorithm along with the map-reduced tools was used to overcome the problem. Santosh [25] proposed artificial intelligence-based solutions for the prediction of the different outbreaks of the epidemic across the world. It will help to identify COVID-19 outbreaks as well as forecast their nature of spread. Bhapkar et al. [26] proposed mathematical modeling to forecast the severity of the pandemic situation by predictive modeling of death tolls. As both the recovery and mortality rate change over the period, thus the authors used progressive recovery rates and progressive mortality rate for the predictive modeling. Dey et al. [27] showed a comparative study of the human mind under a prolonged lockdown period and how people reacted to such time window. Results showed under lockdown people paved their way for their passion in their house (results obtained after analyzing collected tweets), moreover, with the upliftment of the lockdown people slowly faded from their passion to their profession. Analyzing the emotions from the tweets showed most were neutral and a good share went for the worrying category. Mukherjee et al. [28] proposed a CNN architecture, where the network can be trained on both CT scan images and CXR images giving much more training data to learn the model. Their tailored DNN yielded 96.28% accuracy and a mere false negative percentage. Santosh et al. [29] proposed a dynamic model which can include complex parameters into consideration while predicting the epidemic outcome by not just only relying on SIER modeling, where many parameters are overlooked. The author also focused on the data-driven model which can auto tune the parameter value with the recent parametric values automatically. Das et al. [30] in this paper, a custom CNN network namely truncated inception net is proposed to segregate COVID-19 positive CXRs from other non-COVID cases. The proposed architecture achieved an accuracy of 99.96% in classifying COVID-19 positive cases. Mukherjee et al. [31] proposed a light-weighted CNN architecture to classify CXR samples with COVID positive cases. The proposed architecture has been designed with much fewer parametric values as compared to other models. The proposed architecture achieved an accuracy of 99.69%.

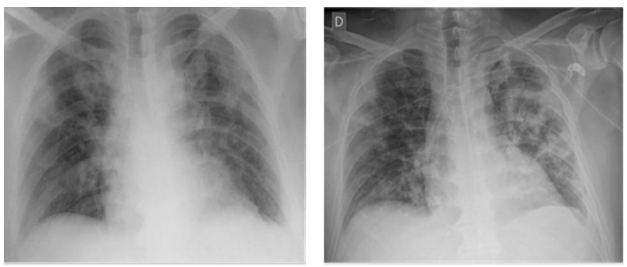
All the works in the CNN classifier model have been concentrated by both using several convolution layers and building the model from the scratch or they have used independent transfer learning models to classify, where they tested on few models and compared the accuracies and chose the best. Most of the work is flawed due to lack of input images and class imbalance for the convolution layer; thus, in our work, we overcame this problem by introducing DCGAN Architecture as a data augmentation tool. Along with that the models shown a tendency of being overfitting and only 1 transfer learning method is not just enough for reliable predictability in such a sensitive case, where the major aim is to minimize the false negative count. The world is going through a pandemic situation and as the new stains of COVID-19 is still not fully interpretable and so search space is very difficult to understand and so here genetic algorithm is an effective way to compute the conditions based on different symptoms. Hereby boosting and grid search method the best parameter is identified, and thus, hyperparameter of the classifier model is tuned. By applying GA operators on conditions the highly correlated factors here, considered as genes are identified and then the classifier model is developed to give the best accuracy and to reduce the number of false negatives cases. If the false negatives are not regulated then it may lead to the worst condition.

Proposed Approach of Ensemble-Based Transfer Learning Model Using Euclidian Weighted Average

In the current situation of pandemic radiological lung imaging of patients plays a key role as the initial screening test. The X-ray predictability can be treated as a preliminary test for COVID-19 patients and only those patients will undergo RT–PCR test if and only if they are being confirmed by X-ray prediction. The patches start developing from day 2 and get significant on day 7, as shown in Fig. 1a, b.

Data Set and Data Pre-processing

In this study, we have considered data sets taken from Open Source authentic organizations. The images are being shared



(a) (b)

Fig. 1 **a** Lung CT image on day 2, showing patchy pattern, ill-defined alveolar condition. **b** Lung CT image on day 7 showing notable patches

as open-source from various radiologists. One of the sources is Cohen JP, and the entire data set is being compiled from few other sources also. Italian Society of Medical and Interventional Radiology (SIRM) COVID-19 DATABASE, Novel Corona Virus 2019 Data set developed by Joseph Paul Cohen, images extracted from different publications [32]. In this study, we have taken 784 training images and 278 validation images and to knockout the class imbalance problem we have augmented images using DCGAN architecture. All the training images are being scaled to (224 × 224) size as those pre-trained models are trained in such dimensions [33, 34].

Noise Removal

The CT images came with some minimal noise and to increase the model accuracy we have tried to minimize the noise. This is one of the steps of image quality being enhanced by applying noise removal methodology. We have used the average filtering method to reduce the noise. The transformation is shown in Fig. 2.

Shadow Removal

The images often contain some shadows due to the availability and position of illumination. In this step, we have tried to segregate the shadow component from the original images (Fig. 3). In this step, we have used Canny Edge detection to remove the shadows. [13, 35]

Image Enhancement

In this pre-processing step, we aim to adjust and tune the color feature of the image. Here we have used an adaptive histogram equalizer to enhance the picture and contrast quality of the image by processing the histogram curve (Fig. 4).

Data Augmentation Using DCGAN Architecture

In the current work, the situation is extremely dynamic and changing every day and the data set is pretty small for accurate prediction of CT image class. Thus, we have augmented data to minimize the class imbalance problem and minimize the overfitting of models.

GAN works on the zero-sum principle and has 2 blocks to generate the synthetic image namely generator and discriminator.

- Generator: the generator or the Generative neural network is mainly responsible for creating a synthetic image to get undetected. It generates the image without training the features of the image of the input data set, i.e., without learning the semantics of the input image data. The loss equation for the generator function is given in the following equation:

$$E_{z\sim p_z(z)}[\log(1 - D(G(z)))]. \quad (1)$$

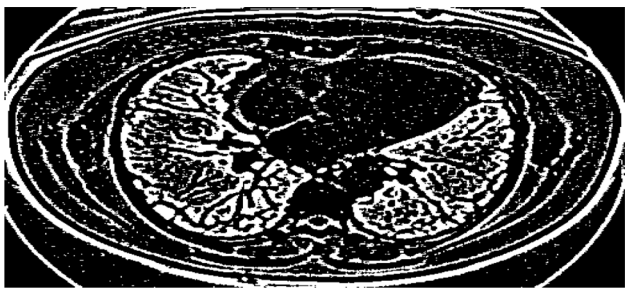


Fig. 2 CT image after noise removal

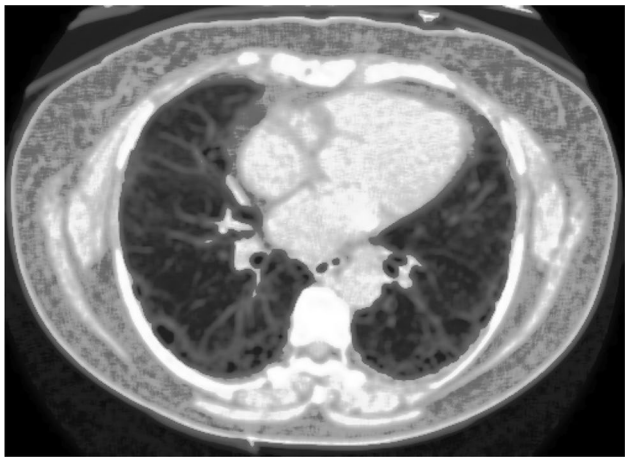


Fig. 3 CT image after shadow removal

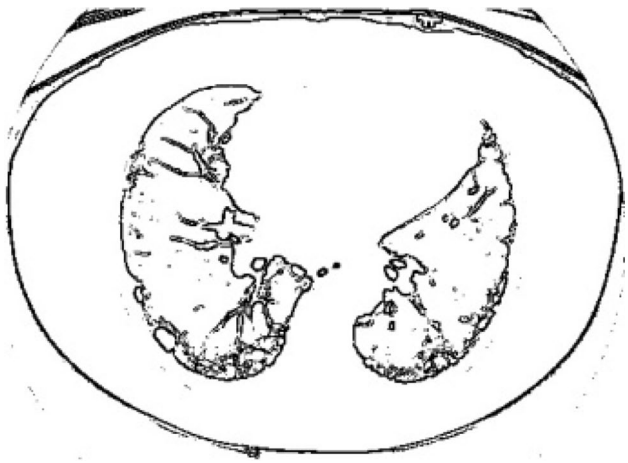


Fig. 4 CT image after applying histogram equalizer

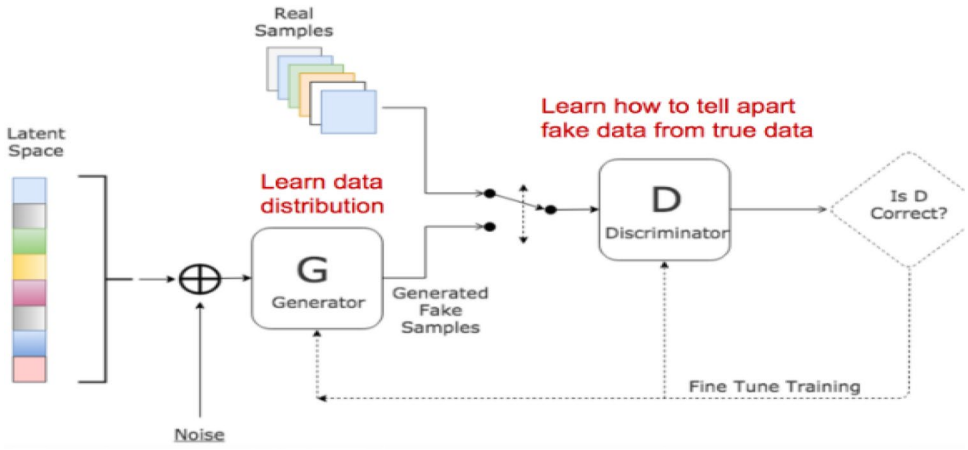


Fig. 5 DCGAN architecture [12]

- Discriminator: the discriminator neural network, learns to classify that the given sample is from the same data distribution or not (Fig. 5). The major goal of a discriminator network is to detect the fake content in the set. It is a classifier network that classifies whether the image is real or not. The loss equation for the discriminator function is given in the following equation:

$$E_{x\sim P_r(x)}[\log D(x)]. \tag{2}$$

The combined loss function for the entire model is shown in Eq. 5:

$$L\big(G, D_o(x)\big) = \int_x (P_r(x)\log\big(D_o(x)\big) + P_g(x)\log\big(1 - D_o(x)\big))\mathrm{d}x, \tag{3}$$

$$L\big(G, D_o(x)\big) = \log\frac{1}{2}\int_x P_r(x)\mathrm{d}x + \log\frac{1}{2}\int_x P_g(x)\mathrm{d}x, \tag{4}$$

$$L\big(G, D_o(x)\big) = -2\log 2. \tag{5}$$

The images formed using DCGAN architecture are shown in Table 1.

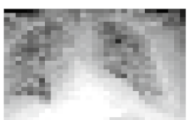
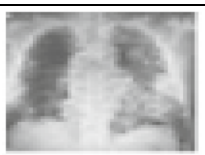
Proposed Algorithm

The pre-trained state-of-the-art CNN classifier model is being tested at first with classifying the images into 2 major classes namely, COVID and non-COVID. The pre-trained models were used namely, ResNet50, VGG16, VGG19, Xception, and InceptionV3. Thus, a combined model always gives an upper hand in terms of accuracy and prediction. Thus, in this section, we have proposed a weighted Euclidian average method (WEAM) for higher model accuracy.

At first, all the models are being used independently, and afterward, they have been combined as an ensemble model. It is expected that the ensembled model will provide a more robust prediction. We have chosen the ResNet50 model as it is one of the contemporary models that are being used with fewer parameters with high accuracy. This model in particular is easy to train and converges fast. Other architectures which are chosen here give us higher accuracy with the considerable low amount of parameters, thus making it faster. For the InceptionV3 model, there are 11 inception layers present in the architecture.

The salient feature of our proposed model is to make a weighted average network, where the providing equation is based on weighted Euclidean average method. Let say our

Table 1 Synthetic CT lung images created using DCGAN Architecture for 50 and 300 Epochs. Source: Created by Author, based on the dataset

$Image\ Size = 128 * 128$ $Noise\ Size = 10$ $Discriminator\ Learning\ Rate = 0.00034$ $Generator\ Learning\ Rate = 0.00036$ ***After* 50 *Epochs***.	
After 300 Epochs with a constant parameter	

VGG19 model is working better than the other models i.e., it has a lower validation error in comparison to the other models which is being there. This implies that it assigned weights better to the classes better than the other models [36].

Assuming the accuracy for a kth model is being q_1.

Therefore, the validation accuracy error is $(100 - q_1)$.

Thus, we define a new weighted factor:

$$d_i = (100 - q_1). \tag{6}$$

All the models are assigned $(1, 1, 1, 1, 1)$ value initially, signifying all the models hold equal weightage and importance:

$$D = \sum (d_i^2 + 0.01 d_{i+1}^2 + 0.01 d_{i+2}^2 + 0.01 d_{i+3}^2 + 0.01 d_{i+4}^2), \tag{7}$$

$$p_i = \frac{d_i^2}{D}. \tag{8}$$

Therefore, weight for the ith network is given as

$$t_i = 1 - \sqrt{\frac{\left(1 - d_i^2\right)^2 + \left(1 - d_{i+1}^2\right)^2 + \left(1 - d_{i+2}^2\right)^2 + \left(1 - d_{i+3}^2\right)^2 + \left(1 - d_{i+4}^2\right)^2)]}{5}}, \tag{9}$$

$$w_i = \frac{\frac{1}{t_i^2}}{\sum \frac{1}{t_i^2}}. \tag{10}$$

Now, let us assume that the output probabilities for class 1 and class 0 are in the form of $[x_0, x_1]$. Now, similarly, the predictive output probability for all the 5 networks be, $[x_{01}, x_{11}]$, $[x_{02}, x_{12}]$, $[x_{03}, x_{13}]$, $[x_{04}, x_{14}]$, $[x_{05}, x_{15}]$ for the models 1, 2, 3, 4, 5, respectively.

Now, let the weight for the models are, respectively, for models be w_1, w_2, w_3, w_4, w_5 calculated from Eq. 10. Therefore, average weight A is calculated using Eq. 11 [37]:

$$A = \frac{\{(w_1 \times x_{01}) + (w_2 \times x_{02}) + (w_3 \times x_{03}) + (w_4 \times x_{04}) + (w_5 \times x_{05})\}}{(w_1 + w_2 + w_3 + w_4 + w_5)}, \frac{\{(w_1 \times x_{11}) + (w_2 \times x_{12}) + (w_3 \times x_{13}) + (w_4 \times x_{14}) + (w_5 \times x_{15})\}}{(w_1 + w_2 + w_3 + w_4 + w_5)}. \tag{11}$$

Proposed Algorithm
***Input*: Covid − 19 CT Images**
***Output*: Predicted Class of Covid − 19 Images**

Begin

At first, resize the Image to $224 * 224$

Then, Noise Removal Algorithm has been used over the image (Average Filtering Method)

Then, Shadows are removed using the Canny Edge Detection algorithm

Afterward, the Images are enhanced using the Adaptive Histogram Method

To handle the class imbalance problem, we augment the data using DCGAN Architecture

Then, Train the Pre − Trained CNN Architectures Models

Calculate Weightage Average for our Proposed Model from Eq. 6.

Lastly, form the Ensemble Architecture using the calculated values and re − train

Prediction

End

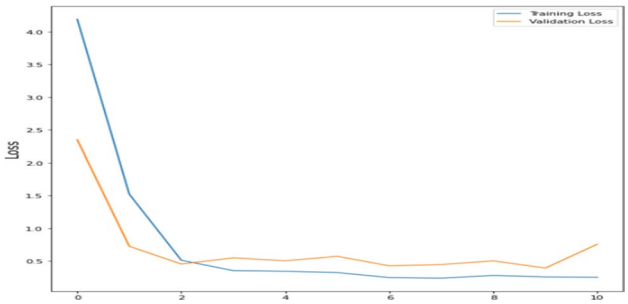


Fig. 6 Training loss curve for ResNet50 architecture

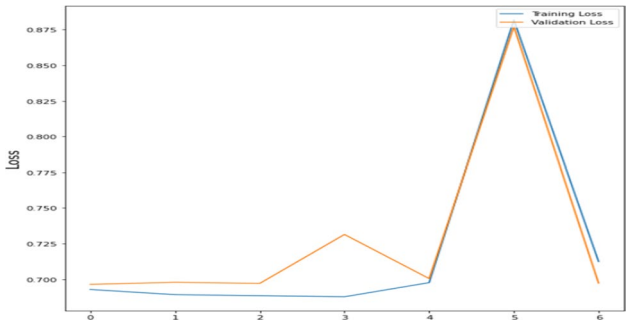


Fig. 7 Training loss curve for VGG16 architecture

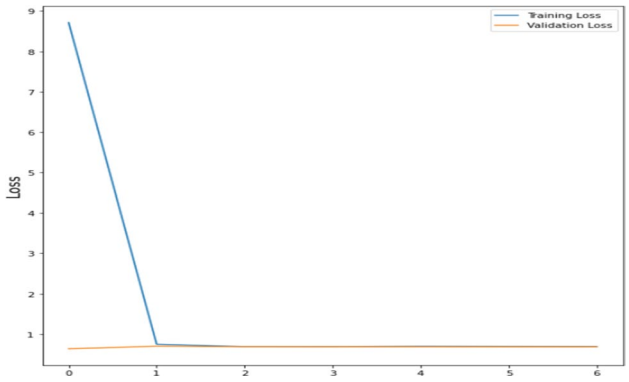


Fig. 8 Training loss curve for VGG19 architecture

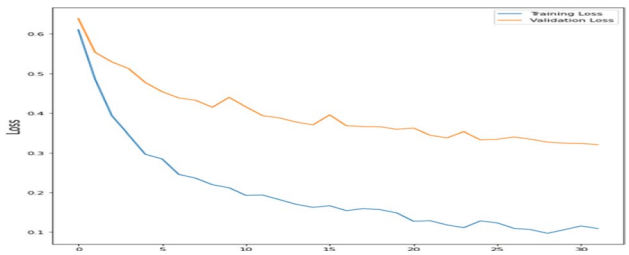


Fig. 9 Training loss curve for Xception architecture

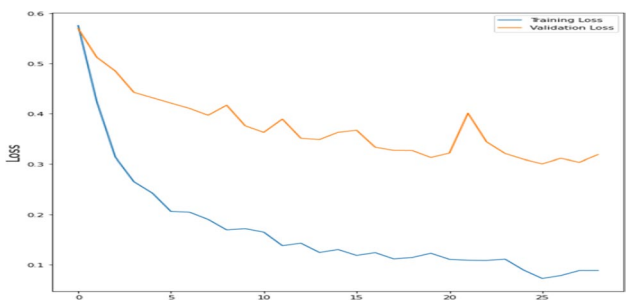


Fig. 10 Training loss curve for InceptionV3 architecture

Results and Discussion

All the models have been trained for 50 epochs on early stopping callbacks being applied so that the epoch could be exited before the model goes overfitting (patience = 8 epochs). We have used Adam optimizer for a faster rate of convergence. Parameters used $\alpha = 0.0001, \beta_1 = 0.9, \beta_2 = 0.889, \in = 1 \times 10^{-8}$.

The training curves for the model are shown in Figs. 6, 7, 8, 9, 10, and 11, respectively [37, 38].

Evaluation Metrics

The evaluation of the proposed model has been done using the standard evaluation metrics available namely, classification accuracy, sensitivity, and F1 score. They are being calculated using Eqs. 12, 13, 14, respectively [39, 40]:

$$\text{Classification accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{12}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{13}$$

$$\text{F1 Score} = \frac{2 \times \text{sensitivity} \times \text{precision}}{\text{sensitivity} + \text{precision}}. \tag{14}$$

Here, TP represents true positive, TN as true negative, FN as false negative, FP as false positive. All the values are being calculated from the confusion matrix of our proposed algorithm.

The comparative results and metrics are shown in Table 2.

Symptom Analysis Using Hybrid Approach of Genetic Algorithm and Classifying Algorithm

In this section, we analyze the symptoms of nCoV affected patients using a hybrid model of genetic algorithm and classifier algorithms. The aim is to find out the level of severity

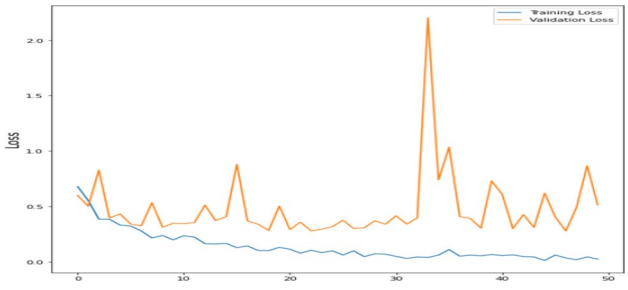


Fig. 11 Training loss curve for proposed architecture

Table 2 Result of pre-trained models and our proposed algorithm. Source: Created by Author, based on the dataset

Models	Parameters	Validation accuracy (%)	Sensitivity (%)	F1 score (%)
ResNet50	25,636,712	85.69	92	94.4
VGG16	138,357,544	92.58	96	92.58
VGG19	143,667,240	88.22	94	93.44
Xception	22,910,480	91.67	91	91.87
InceptionV3	23,851,784	74.44	86	90.55
Proposed Architecture		98.67	99	97.45

among the patients. In the data set which has been collected from an authentic open-source platform, there are 4 levels of severity present viz. none experiencing, low, medium, and high. The input features which has been incorporated in this study are fever, tiredness, dry cough, sore throat, no symptoms, muscle pain, age, nasal congestion, running nose, and gender.

At first, we have analyzed and measured the accuracy of the classifier model applying to the data set and then we have optimized our model using the genetic algorithm to get the optimized data set, and then it was passed through the classifier model and the accuracy of the latter one overshadowed the former one.

Data Set Description

Data set is COVID-19 Symptoms Checker, where different symptoms. It is open-source data. It has 316,800 and 27 columns. The above-mentioned data set consists of anonymous data from an infected person reported positive and admitted to the hospital. Some of the important features in the data set are 'breathing problem', 'fever', 'dry cough', 'sore throat', 'running nose', 'asthma', 'chronic lung disease', 'headache', 'tiredness' etc. This data set will predict whether anyone is having a COVID-19 or not depending on symptoms. Other features include Experience of symptoms like nasal congestion, runny nose, and severity, etc.

In the data set COVID-19 exposure: the exposures of symptoms are considered. For visualization and understanding the risk, pre-processing the raw data having columns like date of the test, date of the visit have been reduced to exposure period, diagnose period, days elapsed from showing symptoms, days are taken for visiting the hospital are considered. It can be illustrated in Fig. 12.

Bayesian Modeling and Feature Visualization for GA Feature Selection

In the last decades, the implementations of the Bayesian hierarchical model (BHM) gained a lot of ground in Artificial Intelligence. Applying this prediction of uncertainty can be done and it is very helpful in the creation of a framework for risk analysis in the healthcare section. Health is the major wealth of humans. It is an organized representation of probabilistic relationships between input and output. It gives conditional independencies. We have modeled for severity or condition or risks of the patient. We have used the severity class as our target variable to assist the much serious patient in the first treatment. To assess the relationship among the variables with the target variable we plot the Bayesian graph, which is acyclic in nature (Fig. 13).

Classifying Algorithm

One of the key formulation and category of it is the classification. In classification main target is to categorize the data into different classes based on features:

$$(\text{features}) \rightarrow \text{target class.} \tag{15}$$

The model is trained on training sample to say I, where $I = \{(x^1c^1), (x^2c^2), \ldots\}$ where x is the input vector and c is the specified class (Fig. 14). The key features (f_i) that influence categorization is part of input x, where $\in x$ is known as the best features and it is selected based on some scores [41].

So for building a good classifier understanding the redundant features and extraction and selection of the best is a very important part of it. After the removal of redundant or irrelevant features, the accuracy of the model increases and the computation time decreases. Therefore, basically, it is a method, where a subset:

$$Z_m = \{x_i^1, x_i^2, \ldots.\} \text{where} 0 < i \leq m$$

where $m < n$($n$is total no of input features),

select/vectors $X\{x_1, x_2, \ldots .x_n\}$where n is the total length of the input vector to optimize the objective function. Now, there are mainly divided into three categories of feature selection filters, wrappers, and embedded selectors.

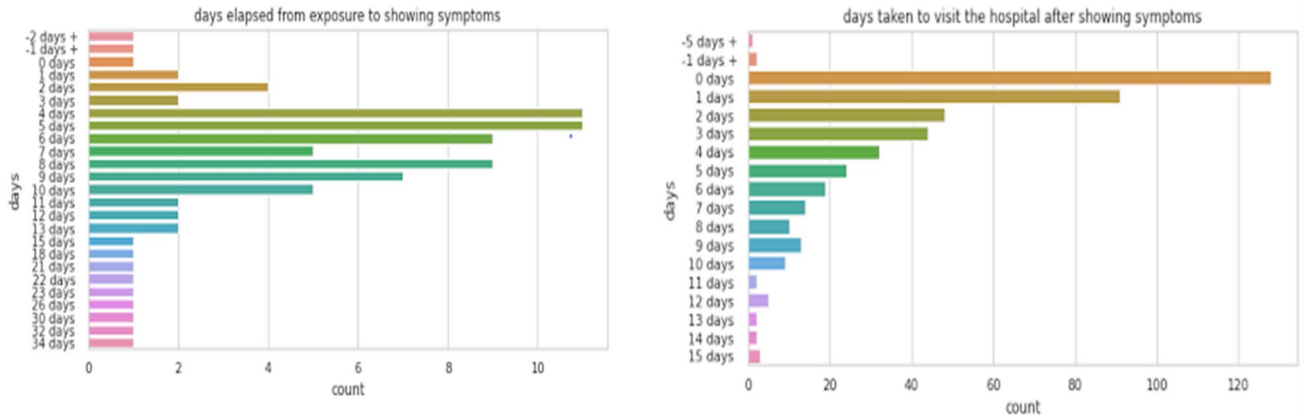


Fig. 12 Graph showed days elapsed vs. showing symptom and hospital admission

Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised machine learning algorithm that is applicable for both regressions rather than logistic regression and classification. It makes a non-probabilistic binary discriminative linear classifier. It creates hyperplanes in infinite dimensions and output or prediction is done based on the optimal hyperplane. Kernel function (fx, y) is defined. In higher dimension hyperplane considered as a bunch of observable points whose scalar product is constant. Therefore, for higher dimension, $k(x, y)$ modifies to $k(x_i, y)$ and so for the hyperplane, we can say that

$$\sum_{i=0}^{n} \beta \cdot k(x_i, y) = C. \tag{16}$$

A higher value of k is preferable. In addition, the functional margin has to be maintained. If there are n points then $w.x - b = 0$ is the equation of hyperplane, where w is the normal vector and $\frac{b}{w}$ is the offset of the hyperplane. [42]

Random Forest

The Random Forest model works on the count of impurity and the split occurs in the direction, where the impurity is least. The count of impurity is measured by 2 factors namely Gini index and entropy. Entropy is described as the amount of information needed to correctly describe a sample. A homogenous sample will give o entropy, while heterogeneous will yield 1. It is calculated as represented in Eq. 8. Similarly, the Gini index is measured using inequality in the sample. Gini index value 0 denotes that the sample is

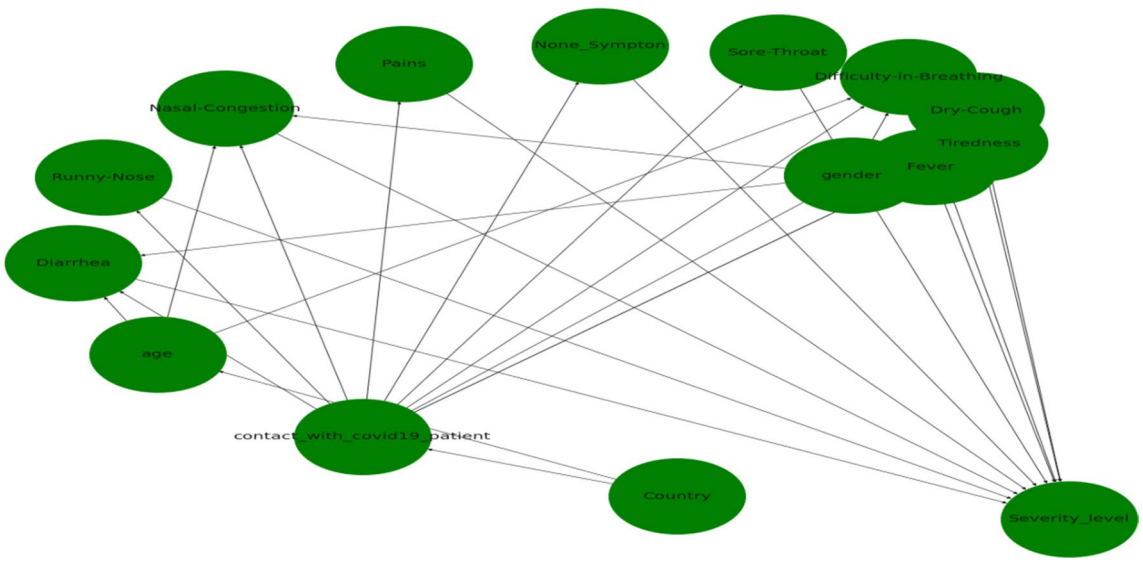


Fig. 13 Bayesian plot for feature relation

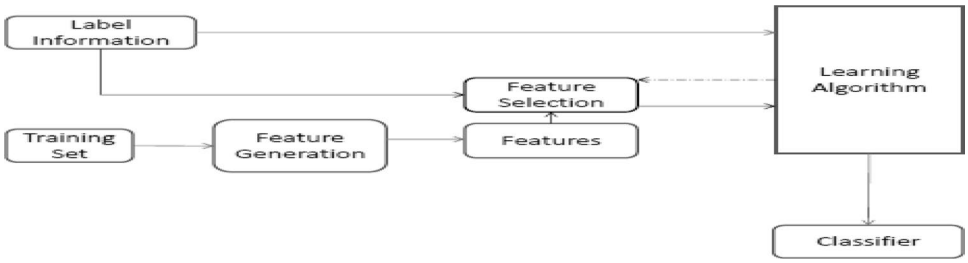


Fig. 14 Framework of feature selection. Source: Jiliang Tang, Salem Alelyani and Huan Liu Feature Selection for

perfectly homogenous and 1 shows it has maximal inequality. It is calculated as represented by Eq. 17:

$$\text{Entropy} = -\sum_{i=1}^{n} p_i \times \log\left(p_i\right), \tag{17}$$

$$\text{Gini index} = 1 - \sum_{i=1}^{n} p_i^2. \tag{18}$$

Naïve Bayes

Naïve Bayes algorithm gives us the probability of a point x which can belong to a specified class C. The probability is calculated based on conditional probability model $P(x_i|C_K)$. The classification involves assigning the value to a class; thus, the proportion can be found using Eq. 19 [43]:

$$p\left(C_a\right)\prod_{i=1}^{n} p\left(x_i|C_a\right) > p\left(C_b\right)\prod_{i=1}^{n} p\left(x_i|C_b\right) \Rightarrow p\left(x_1, x_2 \ldots, x_n|C_a\right) > p\left(x_1, x_2 \ldots, x_n|C_b\right). \tag{19}$$

Thus, the class can be found by mathematical notation given in Eq. 20:

$$C = \arg\max p\left(C_k\right)\prod_{i=1}^{n} p\left(x_i|C_K\right), k \in \{1, 2, \ldots, k\}. \tag{20}$$

Genetic Algorithm

One key factor in optimizing the function and evaluation of the fitness/accuracy of the model and genetic algorithm is an evolutionary, heuristic, domain-independent algorithm widely used for feature selection and optimization. The algorithm is inspired by Darwin's theory of genetics. It is well-suited for multi-featured optimization.

At first, it initializes the randomly generated population. The commencement of the algorithm is from the generation of a population and selection of optimal individuals by measuring objective function and evaluate the fitness function, which takes into account about it fitness in the environment and iterates the process, till the convergence condition is established. In other words, individuals represent one of the solutions to the problem and a set of chromosomes from the population, and the best solution is evaluated using fitness function and the best fitted is considered.

Initialization and Chromosome Encoding

The first step is to define the population and the algorithm changes populations of *chromosomes* (Fig. 15). They are the portrayal of the solution to the problem in string format and a *locus* or a specified position is known as *gene* and the alphabet at that place is *allele*. In GA, we prefer encoding in {0,1}.

Fitness Function and Selection of Features

Assessment of the nature of the chromosome is done and according to criteria it is checked that whether it can be considered as a solution or not. The key selection building block in this algorithm is such that it can act as a guide to the evolution of the Chromosomes and so recombination is also done and having. Higher the rank/score/value more is the chance of getting selected. Some of the methods are Roulette Wheel, Random Stochastic Selection, and Truncation Selection, etc.

Recombination

The 2 major parts of recombination are crossover and mutation. Crossover is the exchange of genes, where the chromosomes are of 2 parents and mutation refers to all possible combinations (swapping and other) of the allele. After this results are fitted to the successor population (Fig. 16).

Evolution

The GA algorithm iterates until and unless the stopping criteria are being reached. After recombination, a new

generation is being created and it also undergoes a similar process to evolve (Table 3). A widely used evolutionary technique is used called replacement-with-elitism. Here there is an almost complete replacement of the wide population in the successor population; this model ensures that the highest fitness does not get lost in the next generation [44].

Along with the feature selection, another important paradigm to increase the model performance is to optimize the hypermeters of the algorithms, and for that, we have used Grid Search Algorithm.

Grid search takes n equally spaced points considering each interval $[a_i, b_i]$ including a_i and b_i. Therefore, the total of n^m possible grid points is possible and it is calculated. Later,

Proposed Algorithm

```
Input: Symptom of nCoV Patients

Output: Severity of nCOV Patients

Begin
    All feature of the dataset is being passed through Genetic Algorithm
     if (Fitness Function of a Generation ! = Maximum Fitness Function)
      {
                for (Iterates through the consecutive generation for max fitness function)
                  {
                  Next Generation  = Max Fitness Function set of Previous Generation
                       Crossover Occurs between maximum fit parents to create Next Generation
                  Next Generation being Created after Crossover
                  }
         break;
        The dataset has been evolved with features giving maximum fitness function value
        Afterward, the evolved dataset has been through different classifier algorithms
        Hyper Parameter tuning using Grid Search Algorithm for the Classifier Algorithms
        Result
      }
End
```

Results and Discussion

We have implemented a hybrid model, at first we have implemented the genetic algorithm-based feature optimization and the iteration is for 10 Generations (Figs. 17, 18). There are 7 parameters or independent variables; thus, the starting generation varied from [0, 0, 0, 0, 0, 0, 0]to[1, 1, 1, 1, 1, 1, 1]. The optimized parameter set is for the feature set given below [45].

The features tabulated in Table 4 give us the features which contribute most to the COVID-19 positive cases.

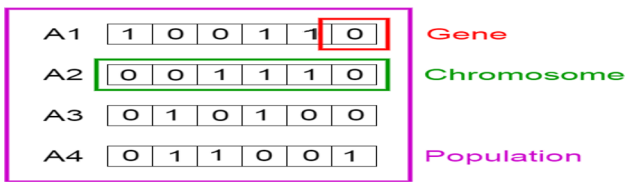


Fig. 15 Showing key components [22]

points (pair) are considered, the maximum of these values is chosen and the best with the highest value is returned. To overcome the overfitting ensemble of the Grid search, stratified cross validation is implemented, where the division of k-folds is done. Let lower bounds $a = (a_1, a_2, \ldots, a_m)$ and a vector of upper bounds $b = (b_1, b_2, \ldots, b_m)$ for each component of ν, where $\nu = (\nu_1, \nu_2, \ldots \nu_m)$ where target is to maximise p value.

The parametric values which are fed into the Grid Search model to get the optimized value are tabulated in Table 5.

Table 6 gives us the results of our model, where the optimized values of hyperparameters are given along with a comparative result of models in where one has been

Table 3 Genetic algorithm result for feature optimization. Source: Created by Author, based on Dataset

Feature Subset	Validation accuracy	Percentile
[1,0,0,1,1,0,1]	84.56%	1

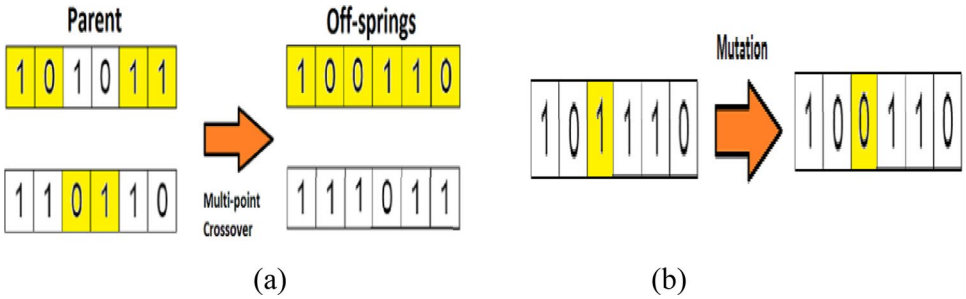


Fig. 16 **a** Represent crossover and **b** represent mutation

gen	nevals	avg	std	min	max
0	100	0.0553085	0	0.0553085	0.0553085
1	67	0.0553085	0	0.0553085	0.0547822
2	56	0.0553085	0	0.0553085	0.0569872
3	64	0.0553085	0	0.0553085	0.0512587
4	60	0.0553085	0	0.0553085	0.0521485
5	59	0.0553085	0	0.0553085	0.0553085
6	64	0.0553085	0	0.0553085	0.0553085
7	52	0.0553085	0	0.0553085	0.0553085
8	49	0.0553085	0	0.0553085	0.0553085
9	57	0.0553085	0	0.0553085	0.0553085
10	52	0.0553085	0	0.0553085	0.0553085

Fig. 17 Genetic algorithm result for 10 generation iteration. Source: Created by Author, based on Dataset

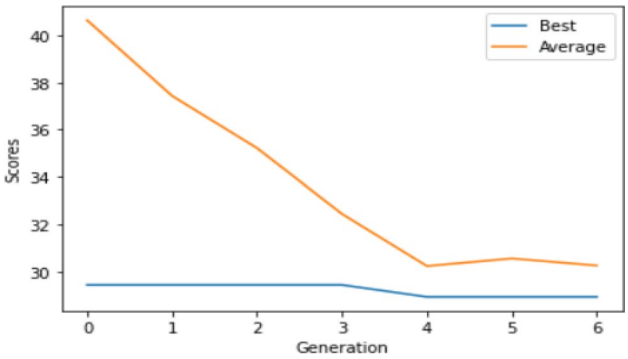


Fig. 18 Genetic algorithm curve across different generations. Source: Created by Author, based on Dataset

Table 4 Features deduced after the data set gone through genetic algorithm. Source: Created by Author, based on Dataset

Optimized features using genetic algorithm
Fever
Difficulty in breathing
Sore throat
Tiredness
Dry cough
Nasal congestion

Table 5 Grid search model parameters

Classifier Model	HYPER-PARAM
Random Forest	'n_estimators': [100, 200, 300, 400,500,600,700,800], 'max_features': ['auto', 'sqrt', 'log2'], 'max_depth': [4,5,6,7,8,9,10], 'criterion':['gini', 'entropy']
SVM	'c': [0.1,1, 10, 100,100], 'gamma': [1,0.1,0.01,0.001,0.0001], 'kernel': ['linear','rbf', 'poly', 'sigmoid']
Naïve Bayes	'var_smoothing': np.logspace(0,-9, num = 100)

optimized after using the genetic algorithm as feature selection and the other is where the classifier model acted without any feature selection.

Concluding Remark and Future Scope of Study

In this study, we have considered a bi-folded study of classifying the images using a proposed weighted transfer learning methodology. The Euclidean Weighted Average is used as the selection equation in the proposed model. As a result, our proposed model gave a high accuracy of 98.67%. Our

Table 6 Accuracy table of optimized and non-optimized parameter. Source: Created by Author, based on Dataset

Classifier model	Validation accuracy (%)	Sensitivity (%)	Parameters optimized using grid search algorithm
SVM	73.52	68	'c': 0.1 'gama':1 'kernel': rbf
Random forest	77.43	70	'n_estimators': 100 'max_features': 'auto' 'max_depth': 4 'criterion': 'gini'
Naïve Bayes	70.66	74	'mean_score_time': array([0.00097208]), 'mean_test_score': array([0.20824808])
SVM [optimized using genetic algorithm]	84.64	79	'c': 0.1 'gama':1 'kernel': rbf
Random forest [optimized using genetic algorithm]	88.96	84	'n_estimators': 100 'max_features': 'auto' 'max_depth': 4 'criterion': 'gini'
Naïve Bayes [optimized using genetic algorithm]	82.36	81	'mean_score_time': array([0.00097208]), 'mean_test_score': array([0.20824808])

proposed algorithm outperformed the traditional models in both terms of validation accuracy and sensitivity. The class imbalance problem has been handled using adding more images and it has data augmented using DCGAN architecture. All the images are pre-processed by removing shadows, noise, etc. for higher classifying accuracy.

In the next section, we have proposed a dual-stage classifier to classify patients into the risk category of low, medium, and high based on the symptoms they are showing. To optimize the result we have used genetic algorithm and the classifier algorithm which are being used are SVM, Naïve Bayes, and Random Forest Classifier. The optimized model has a much higher accuracy which is being expected as 88.96%. Based on this model we can assess and analyze the risk associated with a patient with nCoV symptoms.

The work can be extended using introducing a voice classifier model by which the patient can be classified by the sound of his or her cough. Moreover, other nature inspired algorithms could be incorporated like PSO, Artificial Ant Colony, etc. [46, 47].

Declarations

Conflict of Interest The authors declare that there is no conflict of interest.

References

1. McCall J. Genetic algorithms for modelling and optimisation. J Comput Appl Math. 2005;184(1):205–22. https://doi.org/10.1016/j.cam.2004.07.034.
2. KimH, Park C, Yang H, Sim K. Genetic algorithm based feature selection method development for pattern recognition. In: SICE-ICASE international joint conference, Busan. 2006. p. 1020–25. https://doi.org/10.1109/SICE.2006.315742
3. Rubin M, Stein O, Turko NA, Nygate Y, Roitshtain D, Karako L, Shaked NT, et al. TOP-GAN: stain-free Covid-19 cells cell classification using deep learning with a small training set. Med Image Anal. 2019;57:176–85.
4. Sivakumar S, Chandrasekar DC. Feature selection using genetic algorithm with mutual information. 2014.
5. Babatunde O, Armstrong L, Leng J, Diepeveen D. A genetic algorithm-based feature selection. Int J Electron Commun Comput Eng. 2014;5:889–905.
6. Osisanwo FY, Akinsola JET, Awodele O, Hinmikaiye JO, Olakanmi O, Akinjobi J. Supervised machine learning algorithms: classification and comparison. Int J Comput Trends Technol (IJCTT) 2017;48(3):128–38. www.ijcttjournal.org/
7. DoraisamiS, Golzari S. A study on feature selection and classification techniques for automatic genre classification of traditional Malay music, content-based retrieval, categorization and similarity. 2008.
8. Guyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res. 2003;3:1157–82.
9. Karabulut EM, Özel SA, İbrikçi T. A comparative study on the effect of feature selection on classification accuracy. Proc Technol. 2012;1:323–7. https://doi.org/10.1016/j.protcy.2012.02.068.
10. Guo Z, Uhrig R. Using genetic algorithms to select inputs for neural networks. In: Proceedings of COGANN'92. 1992. p. 223–34.
11. Yang J, Honavar V. Feature subset selection using a genetic algorithm. In: Liu H, Motoda H, editors. Feature extraction, construction and selection. The Springer International Series in Engineering and Computer Science, vol 453. Boston: Springer; 1998. https://doi.org/10.1007/978-1-4615-5725-8_8.
12. Rückstieß T, Osendorfer C, van der Smagt P. Sequential feature selection for classification. 2011. https://doi.org/10.1007/978-3-642-25832-9_14.
13. Yang J, Zhao JX, Cao Q, Hao L, Zhou D, Gan Z, Mao ZW, et al. Simultaneously inducing and tracking Covid-19 cells cell metabolism repression by mitochondria-immobilized rhenium (I) complex. ACS Appl Mater Interfaces. 2017;9(16):13900–12.
14. Xie Y, Xing F, Kong X, Su H, Yang L. Beyond classification: structured regression for robust cell detection using convolutional

neural network. In: International conference on medical image computing and computer-assisted intervention. Cham: Springer; 2015. p. 358–65.
15. Kitrungrotsakul T, Iwamoto Y, Han XH, Takemoto S, Yokota H, Ipponjima S, Chen YW, et al. A cascade of CNN and LSTM network with 3D anchors for mitotic cell detection in 4D microscopic image. In: ICASSP 2019–2019 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE; 2019. p. 1239–43.
16. Pan H, Xu Z, Huang J. An effective approach for robust lung Covid-19 cells cell detection. In: International workshop on patch-based techniques in medical imaging. Cham: Springer; 2015. p. 87–94.
17. Hu H, Guan Q, Chen S, Ji Z, Yao L. Detection and recognition for life state of cell Covid-19 cells using two-stage cascade CNNs. In: IEEE/ACM transactions on computational biology and bioinformatics. 2017.
18. Chen T, Chefd'Hotel C. Deep learning based automatic immune cell detection for immunohistochemistry images. In: International workshop on machine learning in medical imaging. Cham: Springer; 2014. p. 17–24.
19. Zhang J, Hu H, Chen S, Huang Y, Guan Q. Covid-19 cells cells detection in phase-contrast microscopy images based on Faster R-CNN. In: 2016 9th international symposium on computational intelligence and design (ISCID), vol 1. IEEE. 2016. p. 363–67.
20. Indolia S, Goswami AK, Mishra SP, Asopa P. Conceptual understanding of convolutional neural network—a deep learning approach. Proc Comput Sci. 2018;132:679–88.
21. Mani NBS, Suri S, Gupta S, Wig JD. Two-phase dynamic contrast-enhanced computed tomography with water-filling method for staging of gastric carcinoma. Clin Imaging. 2001;25(1):38–43.
22. Lee DH, Seo TS, Ko YT. Spiral CT of the gastric carcinoma: staging and enhancement pattern. Clin Imaging. 2001;25(1):32–7.
23. Thuy MBH, Hoang VT. Fusing of deep learning, transfer learning and GAN for breast Covid-19 cells histopathological image classification. In: International conference on computer science, applied mathematics and applications. Cham: Springer; 2019. p. 255–66.
24. Santosh KC. AI-driven tools for coronavirus outbreak: need of active learning and cross-population train/test models on multitudinal/multimodal data. J Med Syst. 2020;44:93. https://doi.org/10.1007/s10916-020-01562-1.
25. Bhapkar HR, Mahalle PN, Dey N, et al. Revisited COVID-19 mortality and recovery rates: are we missing recovery time period? J Med Syst. 2020;44:202. https://doi.org/10.1007/s10916-020-01668-6.
26. Nilanjan D, Rishabh M, Simon JF, Santosh KC, Tan S, González Crespo R. COVID-19: psychological and psychosocial impact, fear, and passion. Digit Gov Res Pract 2020;2(1, Article 3):4. https://doi.org/10.1145/3428088.
27. Mukherjee H, Ghosh S, Dhar A, et al. Deep neural network to detect COVID-19: one architecture for both CT scans and chest X-rays. Appl Intell. 2020. https://doi.org/10.1007/s10489-020-01943-6.
28. Santosh KC. COVID-19 prediction models and unexploited data. J Med Syst. 2020;44:170. https://doi.org/10.1007/s10916-020-01645-z.
29. Das D, Santosh KC, Pal U. Truncated inception net: COVID-19 outbreak screening using chest X-rays. Phys Eng Sci Med. 2020;43:915–25. https://doi.org/10.1007/s13246-020-00888-x.
30. Mukherjee H, Ghosh S, Dhar A, et al. Shallow convolutional neural network for COVID-19 outbreak screening using chest X-rays. Cogn Comput. 2021. https://doi.org/10.1007/s12559-020-09775-9.
31. https://github.com/ieee8023/covid-chestxray-dataset.
32. Chaudhari P, Agrawal H, Kotecha K. Data augmentation using MG-GAN for improved Covid-19 cells classification on gene expression data. Soft Comput 2019:1–11.
33. Chatterjee A, Roy S. An analytics overview & LSTM-based predictive modeling of Covid-19: a hardheaded look across India. In: Bhattacharyya D, Thirupathi Rao N, editors. Machine intelligence and soft computing. Advances in intelligent systems and computing, vol 1280. Singapore: Springer; 2021. https://doi.org/10.1007/978-981-15-9516-5_25.
34. Han C, Hayashi H, Rundo L, Araki R, Shimoda W, Muramatsu S, Nakayama H, et al. GAN-based synthetic brain MR image generation. In: IEEE 15th international symposium on biomedical imaging (ISBI 2018). IEEE. 2018. p. 734–38.
35. Rashid H, Tanveer MA, Khan HA. Skin lesion classification using GAN based data augmentation. In: 41st annual international conference of the IEEE engineering in medicine and biology society (EMBC). IEEE. 2019. p. 916–19.
36. Salehinejad H, Valaee S, Dowdell T, Colak E, Barfett J. Generalization of deep neural networks for chest pathology classification in x-rays using generative adversarial networks. In: IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE. 2018. p. 990–94.
37. García-Dominguez A, Galván-Tejada CE, Zanella-Calzada LA, Gamboa-Rosales H, Galván-Tejada JI, Celaya-Padilla JM, Luna-García H, Magallanes-Quintanar R. Feature selection using genetic algorithms for the generation of a recognition and classification of children activities model using environmental sound. Mob Inf Syst. 2020. https://doi.org/10.1155/2020/8617430.
38. Saidi R, Ncir WB, Essoussi N. Feature selection using genetic algorithm for big data. In: Hassanien A, Tolba M, Elhoseny M, Mostafa M, editors. The international conference on advanced machine learning technologies and applications (AMLTA2018). AMLTA 2018. Advances in intelligent systems and computing, vol 723. Cham: Springer; 2018. https://doi.org/10.1007/978-3-319-74690-6_35.
39. Prügel-Bennett A, Shapiro JL. An analysis of genetic algorithms using statistical mechanics. Phys Rev Lett. 1994;72(9):1305–9.
40. Jasuja A. Feature selection using diploid genetic algorithm. Ann Data Sci. 2020;7:33–43. https://doi.org/10.1007/s40745-019-00232-5.
41. Hilda GT, Rajalaxmi RR. Effective feature selection for supervised learning using genetic algorithm. In: Electronics and communication systems (ICECS), 2nd international conference IEEE. 2015. p. 909–14.
42. Goldberg DE. Genetic algorithms in search, optimization, and machine learning. Reading: Addison-Wesley; 1989.
43. Olson RS, Moore JH. TPOT: a tree-based pipeline optimization tool for automating machine learning. In: Hutter F, Kotthoff L, Vanschoren J, editors. Automated machine learning. the springer series on challenges in machine learning. Cham: Springer; 2019. https://doi.org/10.1007/978-3-030-05318-5_8.
44. Olson R, Bartley N, Urbanowicz R, Moore J. Evaluation of a tree-based pipeline optimization tool for automating data science. 2016. p. 485–92. https://doi.org/10.1145/2908812.2908918.
45. Bouvrie J. Notes on convolutional neural networks. 2006.
46. Perez L, Wang J. The effectiveness of data augmentation in image classification using deep learning. 2017. arXiv:1712.04621.
47. Chatterjee A, Roy S, Sinha T. Patient arrival to public OPDs: analysis and use of statistical distribution for improving performance indicators in rural hospitals. In: Analyzing data through probabilistic modeling in statistics. IGI Global; 2021. p. 83–114.