Classifying Covid-19 Chest X-ray Images Using Machine Learning Algorithms and Deep Learning: A Comparative Analysis

A. Veronica Nithila Sugirtham[1], Dr. C. Malathy[2]
[1]vn9428@srmist.edu.in, [2]malathyc03@gmail.com
[1,2]Department of Computer Science & Engineering,
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu 603203, India.

Abstract. ***More than a year since the novel coronavirus was first discovered, its presence is still prevalent throughout the world. In such grave situations if technology can help humans combat the spread, then why not explore it. Therefore, keeping this in mind, several researchers have already started investigating artificial intelligence algorithms to find solutions to predict coronavirus using X-ray images of chest. But due to lack of dataset during the initial days of their research, many came up with framework using pre trained image classification models such as VGG-16, Inceptionv3, ResNet-50 and others. In this paper, the performance of two machine learning algorithms which are support vector machine and decision tree has been evaluated. Further developed deep learning model applying convolutional neural network to classify the chest x-ray images as covid-19 or normal. The final CNN model was also integrated with a user interface and hosted on web server for easy access which allows anyone to upload the chest x-ray image from his computer or mobile and check the result.***
Keywords: Coronavirus, Machine Learning, Grey Level Co-Occurrence Matrix, Convolutional Neural Network

1 Introduction

After almost one year of coronavirus disease, dedicated efforts by the scientists and several human trials, vaccination has been finally found for preventing the disease in humans. Though people are stepping forward to get vaccinated, never to ignore the fact that it also has shown some side effects in people such as high-grade fever, body pain, fatigue etc. On one side the effectiveness of vaccine remains a question mark and many people are hesitant to get vaccinated, while on the other side shortage of vaccine were reported in some centres leading to their closure. Despite people following the social distancing of at least 2 arm distance and other hygiene related norms such as regular hand washing and sanitizing of self and other objects used by us, still the graph of people infected each day is a hilly patch with ups and downs. With a sudden surge of positive cases in our country, India stands at third position after United States and Brazil among the most affected countries. The rate of human-to-human transmission around the world and mainly in India has a become a major concern for all authorities including the health officials, government and other controlling bodies. Lockdowns and curfews are being declared by the government in several states of India to restrict the movement and gathering of people, shutting down malls, shops, cinema theatres and other places affecting the normal lifestyle of people, all for the sake of stopping the chain of transmission of the virus.

New strains of the virus are being reported and symptoms more severe directly affecting the lungs instead of the nasopharyngeal region. Symptoms can be mild as in case of flu or more severe including breathlessness and lack of oxygen. Earlier symptoms were fever, sore throat, cough but now severe diarrhoea and body pain are also observed along with the prior ones. The virus can sometimes cause severe infection of the lungs and disturb the normal functioning of the other vital organs causing death of the individual. Case study of patients reveal that the virus adversely affects two sets of people, one is the elderly people above 60 years of age and the other who have underlying health conditions like diabetes, blood pressure, heart disease, kidney disease and lung related diseases. These two sets of people are at higher risk of catching the virus including more severe symptoms as compared to others and a prolonged time of sickness.

Despite having covid testing labs and thermal screeners at place still it seems insufficient to cope with the increasing number of cases. Increasing cases make it difficult for labs to maintain social distancing and lack of kits can sometimes delay the diagnosis. The WHO suggests that breaking the chain of transmission is one of the main solutions to control the spread of virus. Therefore, it emphasizes that testing and isolating the patient as early as possible will majorly contribute to preventing the spread of the virus among the infected patient's family members as well as the society. The current RT-PCR tests require kits to be available for testing and is time consuming due to the lab formalities and processes. Moreover, suspected patients coming for test can have the virus in them and can affect other people around them including the radiologists and technicians. What is then required is a more robust and automated technique to detect coronavirus in patients using X-ray images of chest. Since lungs are the most affected region of the human body therefore using chest radiographs would be a more efficient way in detecting the presence of the virus. Also, labs are nowadays present in all major cities and very accessible by everyone therefore X-ray of chest can be performed easily, and reports are also generated in a fast pace.

The automated method will make the diagnosis process quicker as compared to conventional lab testing followed at present. It will further assist radiologists to be more precise in their decision making who would otherwise have to check the X-ray reports manually which consumes more effort as well as time. Many a time certain minute details can remain unnoticed by the human eyes but with the help of an artificially intelligent system trained to classify the images, can enlighten those areas to radiologists and help them conclude accurately. Artificial intelligence in the field of medicine is progressing rapidly, in aiding doctors in various things like performing minimally invasive surgeries, locating any tumour or stone in the body, and many other examples are there. This advancement of technology, combining software with medicine has proven to be a friend of doctors, guiding them in making the right decisions at the right time and to give the best quality treatment and care to their patients. With shorter period of diagnosis, doctors can start the treatment on time and thus casualty rates can be reduced.

This paper evaluates the model performance using supervised machine learning algorithms as support vector machine, decision tree further moving ahead to convolutional neural network for the optimum performance on classifying chest X-ray images into two classes covid vs.

normal and the final model was integrated into a web app which was then hosted. Supervised machine learning algorithms were selected initially because our dataset contains labelled images belonging to binary classes COVID or NORMAL. By labelled images, we mean that the images are already tagged to the correct class and the model learns from the labelled training data and predicts on the unforeseen data.

2 Literature Survey

Used grey level co-occurrence matrix to extract features from the dataset, total 12 features were extracted (autocorrelation, energy, entropy, contrast, dissimilarity, cluster prominence, cluster shade, correlation, sum of squares variance, sum average, maximum probability and homogeneity). Further used these features to detect bone abnormalities in musculoskeletal radiographs by training LBF SVM classifier achieving 62% accuracy. GLCM are mostly used in image classification as it represents the second order information of the adjacent pixels in an image [1]. In this paper the dataset was collected from Kaggle having 279 images for each covid positive and covid negative classes. Since the dataset was small, image augmentation was done and operations such as horizontal flip, shifting, shear and zoom were performed. Proposed model was a multilayer convolutional neural network with a transfer learning Inception V3 model which extracts features like CNN using convolutional and pooling layers but contains weights of dataset ImageNet. This model achieved validation accuracy of 84% [2]. In this paper, three pre-trained DCNN networks such as NASNet, DenseNet and MobileNet have been used in an ensemble network. The features extracted from each model has been concatenated into a classifier before doing the actual classification. All 3 models were initially trained on ImageNet dataset. The proposed ensemble model achieved an accuracy of 91.99% and classified the X-ray images into three classes namely community acquired pneumonia (CAP), Covid19 and Normal [3]. In this paper, histogram equalization and CLAHE was used to improve the quality of images. Then to extract resources from 708 CTs (312 COVID & 396 NON COVID) basic CNN was used. Multiple classifiers were used on the extracted data for classification between the two binary classes. An accuracy of 97.88% was achieved [4]. In this paper, dataset was collected from public databases present online in various git repositories and a total of 6249 (out of which only 529 belong to covid positive class and rest 5928 were of patients testing negative for the virus) chest radiographs were collected for the study. Pre-trained models such as ResNet and Xception were used for covid 19 classification of chest radiographs. ResNet achieved an accuracy of 82.5% and Xception model achieved an accuracy of 97.4% [5]. In this paper, a fine-tuned CNN (FT-CNN) was used and the outcome was exactness of 90.70%, testing precision of 90.54 % and an accuracy of around 94% on the chest X-shaft pictures [6]. In this paper, two approaches have been taken. The first one consists of evaluating the patient registration slips with real time data using ResNet-101 which achieved an accuracy of 81%. For the second approach, dataset of 8009 chest radiographs were collected and then three neural networks were used, which are faster R-CNN, mask R-CNN and ResNet-50. The first one achieved an accuracy of 87%, second one 83% and finally the least 72% was achieved on ResNet-50 network. [7]. In this proposed model, LeNet-5 CNN architecture was used and total 82,146 parameters were trained on the initial CT frames accounting to 349 covid and 397 non covid CT frames which was then increased using image augmentation technique and

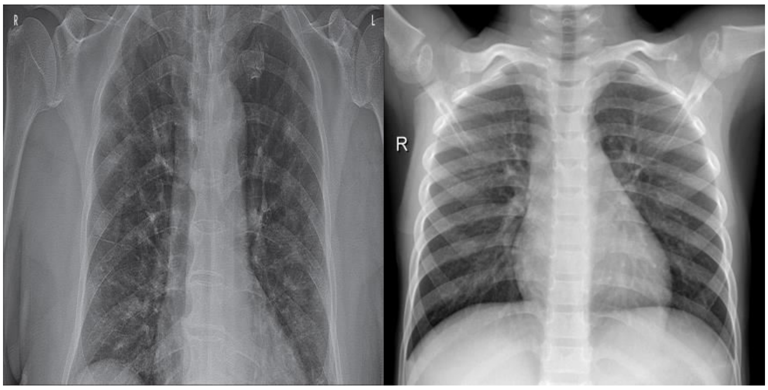
ISSN 2515-8260 Volume 08, Issue 03, 2021

resulted in 1744 covid and 1588 non covid CT frames. The accuracy achieved was around 86.06% [8]. This paper presents a joint model of cycle generative adversarial network (CycleGAN) and the convolutional neural network. The CycleGAN was used to generate more reliable images from a limited dataset and to localise the lesion areas. The model was trained on both chest CT as well as X-ray images dataset. It was observed that their model achieved better performance over others such as GAN-CNN, Fixed-Point GAN and AG-CNN with an accuracy of 95.9% on CT images and 87.6% on chest radiographs [9].

3 Methodology

3.1Dataset

Firstly, dataset was collected from online resource Kaggle (Fig. 1) where images from different hospitals have been consolidated into a repository. In this dataset, there were primarily two classes, one COVID containing 1200 images and the other NORMAL containing 1341 images. These two classes were used to train and test our support vector machine and decision tree classifiers by dividing the dataset into 70:30 ratio. Next for developing the CNN model the dataset was further divided into training set containing 1005 COVID and 1027 NORMAL images, validation set containing 97 COVID and 157 NORMAL images.



(a) COVID (b) NORMAL

Fig. 1. Sample images from the two classes of our dataset

3.2 Feature Extraction

Before proceeding with evaluating the performance of machine learning algorithms first step needed is to extract the features from the images. Since our dataset contains all grey scale images therefore Grey Level Co-occurrence Matrix (GLCM) was used for extracting features. GLCM is most preferred for analysing the second order texture features. Total 6 features were extracted from our dataset of x-ray images which are contrast, dissimilarity, homogeneity, energy, correlation and angular second moment (ASM). Let us assume I co-occurrence matrix with J dimension and m, n are its coefficients and coordinates of the elements. The grey level features are calculated by considering a square matrix based on a Region of Interest (ROI) dimension of the number of grey levels (J) in the X-ray images. Details of each feature are as follows:

- **Contrast:** The local variations present in the image are determined by this feature. The GLCM_Contrast function is defined in the Equation (1) as follows

$$GLCM_{Dissimilarity} = \sum_m \sum_n |\mathrm{m} - \mathrm{n}|^2 \mathrm{p}(\mathrm{m}, \mathrm{n}) \quad (1)$$

- **Dissimilarity:** This feature measures the mean difference in the distribution of the pixels in the images. That is the distance between the pair of objects in the ROI. The GLCM_Dissimilarity function is defined in the Equation (2) as follows

$$GLCM_{Dissimilarity} = \sum_m \sum_n |\mathrm{m} - \mathrm{n}| \mathrm{p}(\mathrm{m}, \mathrm{n}) \quad (2)$$

- **Homogeneity:** This feature measures the homogeneity between the pair of objects in the ROI. The GLCM_Homogeneity function is defined in the Equation (3) as follows

$$GLCM_{Homogeneity} = \sum_m \sum_n \frac{1}{1+|m-n|^2} p(m, n) (3)$$

- **ASM:** The degree of pixel pair repetitions is measured by this feature. In other words, it measures the uniformity of the pixel pairs. The GLCM_ASM function is defined in the Equation (4) as follows

$$GLCM_{ASM} = \sum_{m,n} p(m, n)^2 (4)$$

- **Energy:** $GLCM_{Energy} = \sqrt{GLCM_{ASM}} (5)$

- **Correlation:** This is a measurement of how a pixel is correlated with its neighbouring pixels over the whole image. In other words, it is the linear dependency of grey levels. The GLCM_Correlation function can be defined in the Equation (6) as follows

$$GLCM_{Correlation} = \sum_m \sum_n \frac{(m-\mu_m)(n-\mu_n)p(m,n)}{\sigma_m \sigma_n} (6)$$

Where μ_m and μ_n represents the average on row m or column n and σ_m or σ_n represents the variance on row m or column n.

To implement GLCM in python greycomatrix and grecoprops were imported from skimage. feature.texture module. Since each of the 6 images had an array of 4 values therefore the resulting CSV file had 24 columns and the number of rows is the total of number of images of each class. So, total 2 CSV files were generated one for COVID and the other for NORMAL class which were then used as the dataset for training and testing the machine learning classifiers.

3.3 Machine Learning Classifiers

Two supervised learning algorithms were used for this study. Firstly, support vector machine (SVM) was taken. In SVM algorithm each data item is plotted as a point in the n-dimensional space where *n* is the number of features we have with the value of a particular coordinate represented by the value of each feature. Then classification is performed by finding the hyper-plane which differentiates the two classes very well. The coordinates of individual observation are known as support vectors. The SVM classifier is used to best segregate the two classes and it is available in the scikit-learn library in Python. From sklearn, imported svm, then using the dataset object was created, further the model was trained by splitting the dataset into 70:30 and finally prediction was done on the test data. The learning curve was plotted by importing plot_learning_curves from mlxtend (machine learning extensions), a python library. The misclassification error decreased as the size of the training set kept increasing which can be observed in the graph (Fig. 2). Test accuracy of 0.8938 i.e., approximately 89 % was achieved.

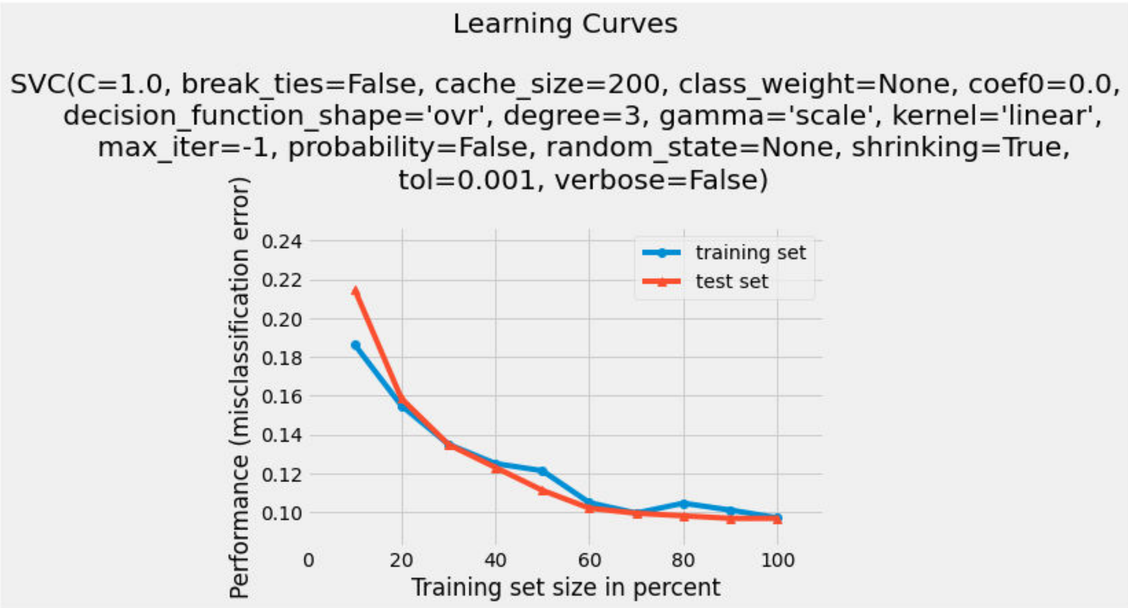


Fig. 2. Learning curve plot for SVM classifier

Next, another supervised machine learning algorithm called Decision Tree was explored to check if there is any chance of increasing the accuracy. The decision trees use a tree representation to solve the problem in which class labels are represented by each lead node and the internal nodes of the tree represent attributes. To use decision tree classifier in python, similar to svm, the classifier was imported from sklearn.tree, then object was created, model was trained on the 70:30 dataset and finally prediction was done on the test data. Similar to svm, the learning curve was plotted using mlxtend.plotting.plot_learning_curves function. The misclassification error of the test set decreases as the size of training set increases which can be observed in the graph (Fig. 3). Test accuracy of 0.9187 i.e., approximately 92 % was achieved which was better than the SVM performance.

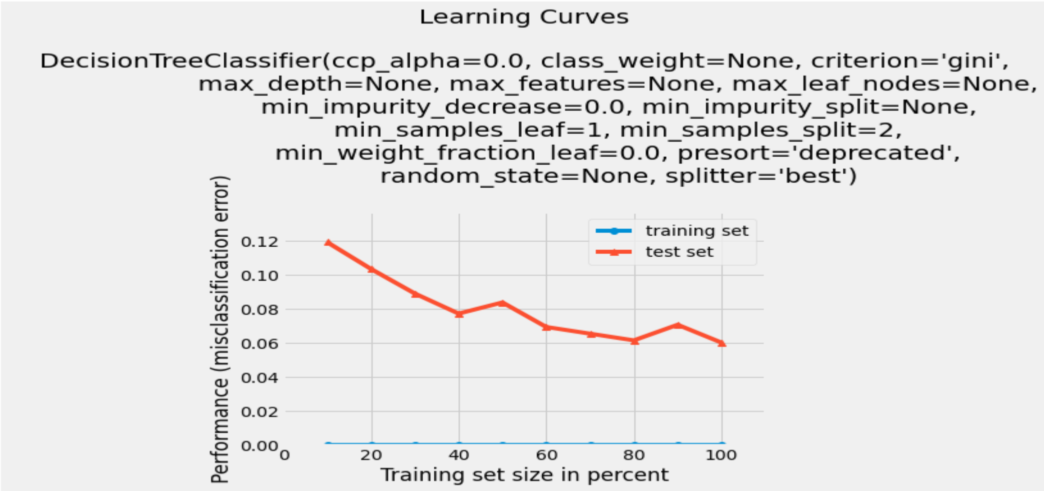


Fig. 3. Learning curve plot for Decision Tree classifier

3.4 Convolutional Neural Network (CNN) model

Though the machine learning algorithms performed well on our dataset there was scope of improvement therefore, the most preferred algorithm for image classification tasks i.e., CNN was selected to be trained from scratch instead of using any pre-trained models. To develop our CNN model, Google Colaboratory also known as Colab was used which is basically Jupiter Notebook only. The dataset prepared was moved inside the google drive. Next, the drive was mounted and directories to the respective folders were set. To visualize the images Matplotlib was used as shown below (Fig. 4)

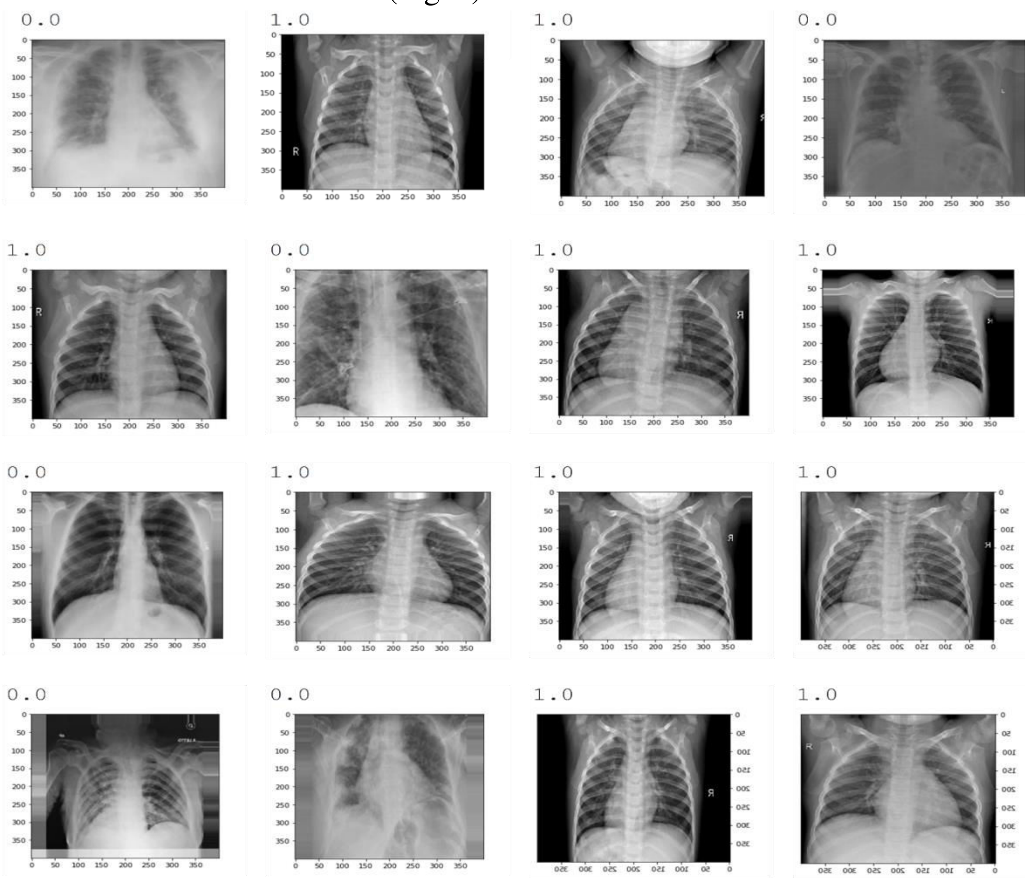


Fig. 4. Image Visualization using Matplotlib

Then began to develop our model. It was initialised using the Keras.Sequential model. The architecture (Fig. 5) of the model contains three convolutional blocks each with a max pooling layer. The first two convolutional blocks were of 32 filter type (number of feature detectors), 3*3 kernel size, input shape of 400*400*3 (input shape will be given only to first layer) and activation function as 'relu' (makes all negative values to zero). The third convolutional block was of 64 filter type with same 3*3 kernel and activation function as relu. The max pooling layer in each convolutional block had a pooling window of 2*2. (used after each convolutional layer for down sampling the feature maps to get a summarized version of the input). Then flatten layer was used to convert the output into one dimensional and sent to the dense layer which is a fully connected layer of 64 units on top of it and is activated by relu function.

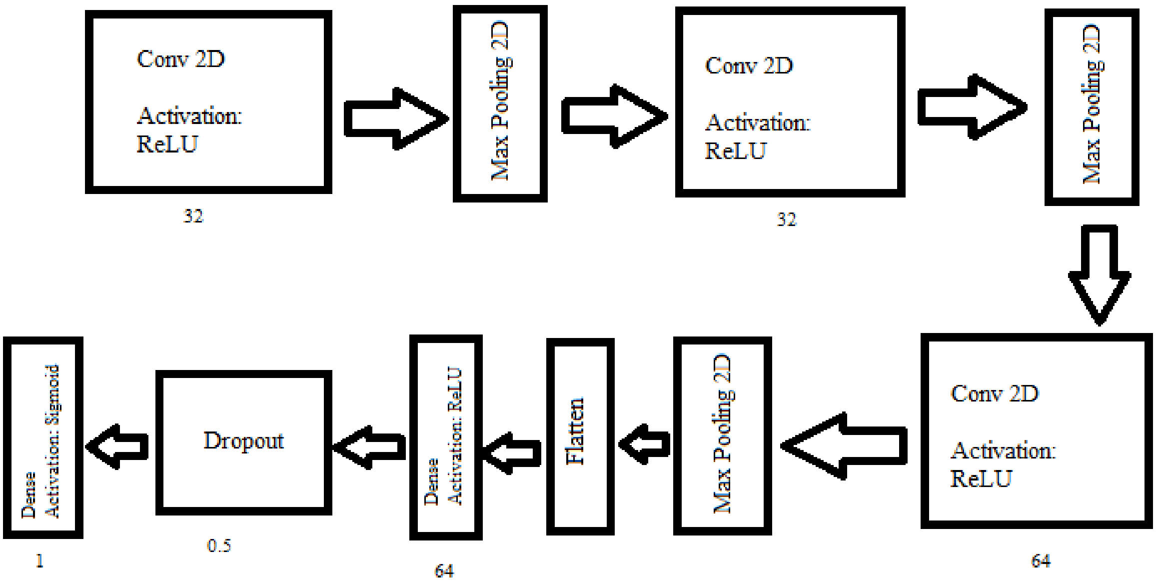


Fig. 5. Block diagram of our model architecture

The model was then compiled using Adam optimizer with a learning rate of 0.0001 and to view the training and validation accuracy in each training epoch metrics argument is passed. Next step was to tarin the model. For this the training and validation batch size were set to 16 and 2 respectively because the number of steps in each epoch = total no. training or validation samples / training or validation batch size. Two scenarios were executed:

1. In one case the number of epochs were set to 5, were training accuracy of 0.9753 and validation accuracy of 0.9646.

2. But, in the second case the number of epochs which is one of the hyperparameters for CNN was experimented and set to a larger number 100 with early stopping, monitoring on val_loss and patience of 5 (that is it will wait till 5 epochs before stopping). This case gave training accuracy of 0.9821 and validation accuracy of 0.9843 (i.e., approximately 98% which is the maximum value as compared with earlier SVM and Decision tree). Hence this was considered as the final BEST FIT model. Patience of 5 was used because

(a) lesser values like patience =1 was stopping after 2 epochs (in case of batch size=32) and
(b) maximum stopping after 4 epochs (in case of batch size = 16).

One more thing was also observed that when the batch size was kept large around 127 then the model resulted in underfitting.

3.5 Integrating user interface (UI) with our model and hosting

As a final step, our CNN model was integrated into a web app which was then hosted, so that it can be easily shared with others and used by them to upload their chest X-ray image to the model and get the result predicted. A simple web app giving users the functionality to upload chest X-ray image on click of a button and getting the result within few seconds was designed and developed using Streamlit, an open-source python library. Finally, to deploy our web app on cloud, Heroku was used. Heroku is a platform-as-a-service (PasS) provider, giving developers a free platform to run and manage web apps.

4 Result

4.1 Training Process

Our model was trained using Google Colab Pro version, providing tensor processing unit (TPU) as hardware accelerator and High RAM as the runtime. We used 1.44 GB RAM and 38.57 GB TPU. Our model was trained for 12 epochs (due to early stopping) taking time around 22 minutes time in total (approximately 1.8 minutes per epoch).

4.2 Model Evaluation

The model performance on our training set and validation set was evaluated after each epoch as shown in Table 1. For evaluation four parameters were considered which are training loss, training accuracy, validation loss and validation accuracy.

Table 1. Loss and accuracy at the end of each epoch

After Epoch	Training Loss	Validation Loss	Training Accuracy	Validation Accuracy
1	0.4164	0.1151	0.7931	0.9803
2	0.1817	0.0617	0.9465	0.9803
3	0.1285	0.0581	0.9528	0.9843
4	0.0958	0.0501	0.9710	0.9803
5	0.0915	0.0446	0.9819	0.9843
6	0.0931	0.0413	0.9691	0.9843
7	0.0692	0.0337	0.9751	0.9843
8	0.0837	0.0519	0.9743	0.9764
9	0.0986	0.0420	0.9773	0.9843
10	0.0824	0.0660	0.9766	0.9803
11	0.0629	0.0475	0.9810	0.9843
12	0.0635	0.0391	0.9812	0.9843

After the training process the training loss was 0.0635 and validation loss was 0.0391 and as can be seen in the graph (Fig. 6) both losses decreased over epochs.

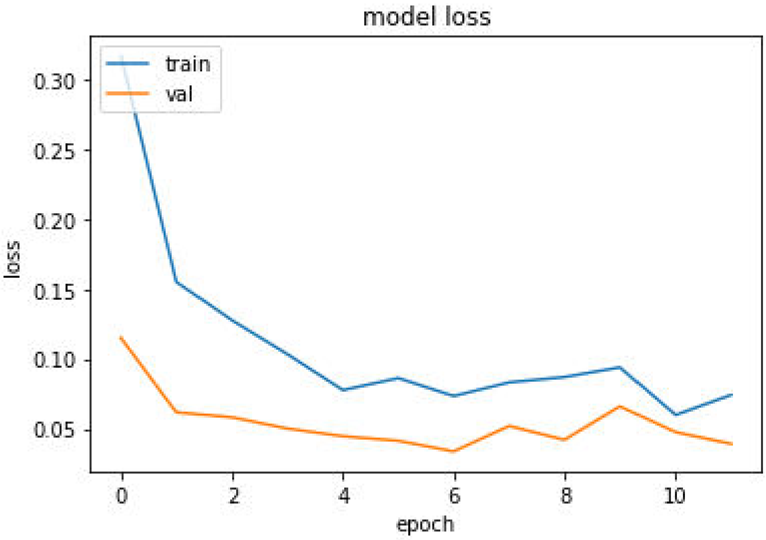


Fig. 6. Model loss vs. Epoch

On the other hand, after the training process, the accuracy was 0.9812 (i.e., 98.12%) on training set and 0.9843 (i.e., 98.43%) on validation set. The accuracy increased over epochs as can be observed in the graph (Fig. 7). In the whole, our model gave a good performance on the dataset collected.

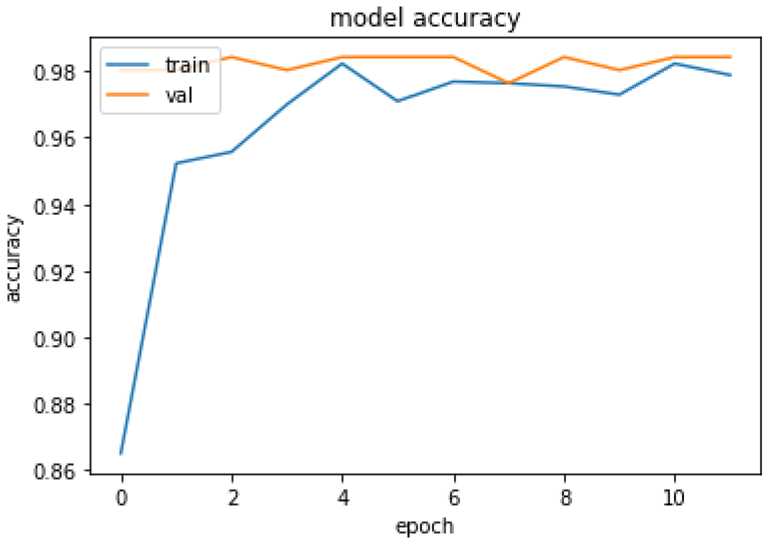


Fig. 7. Model accuracy vs. Epoch

4.3 Comparison

Our literature survey covered different models which have already been developed including information such as how much dataset was collected, whether any image augmentation method was used, which model was used and how much was the accuracy at the end of training process. The comparison of those models vs our model performance is shown in Table 2. It can be clearly seen that most of the existing covid classification architecture used pre-trained models for image classification such as ResNet-50, Inception V3, LeNet-5, MobileNet, DenseNet, NASNet, Xception and others. The best performing among all was the one in which CNN was used to extract features and then multiple classifiers such as XGBoost, random forest, and multilayer perceptron classifiers were used for classification giving an accuracy of 97.88% [4]. So, our model gave better performance than all the existing models mentioned in the literature survey by achieving an accuracy of 98.43%.

Table 2. Comparing the performance of different models

Model	Details	Accuracy
ResNet50 [5]	For improving learning accuracy of the model 50 convolutional layers with skip connections are used	82.5 %
Mask R-CNN [7]	Given an image, it can separate it into classes, masks and object bounding classes	83 %
CNN with Inception V3 [2]	Dataset was small. Trained for 31 epochs. No evident overfitting.	84 %
LeNet-5 CNN architecture [8]	Used open-source dataset containing 349 covid CT of lungs and 397 non covid CT images. Total number of parameters trained were 82,146	86.06 %
Faster R-CNN [7]	Object detection architecture that uses convolutional neural networks like single shot detector (SSD) and you look only once (YOLO)	87 %
CycleGAN and CNN joint model [9]	CycleGAN was used to generate more reliable images from a limited dataset and to localise the lesion areas	87.6 %
Ensemble of NASNet, MobileNet and DenseNet [3]	The features extracted from the 3 algorithms were concatenated before classification which were initially trained on ImageNet dataset	91.99 %
FT-CNN [6]	CNN model with fine tuning	94 %
Xception [5]	Improved inception model with modules been replaced with depth wise convolutions.	97.4 %
CNN to extract features and multiple classifiers for classification [4]	Basic CNN was used to extract resources from 708 CTs. Then XGBoost, random forest, and multilayer perceptron classifiers were used for classification	97.88 %
Our Proposed CNN model	Dataset contained 1200 covid and 1341 normal chest radiographs. Trained for 12 epochs. No evident overfitting.	98.43 %

5 Conclusion

In this paper, a convolutional neural network model was proposed to be developed from the scratch without using any pre-trained models. This model performed exceptionally well on our dataset collected from Kaggle. The training accuracy achieved was 98.12% and validation accuracy achieved was 98.43% outperforming all the other models discussed in the survey. The main reason is that initially when researchers had started their work dataset was limited and not many covid positive chest radiographs were available. But, this proposed work was carried out during the peak moment of the pandemic and therefore we could collect good quality without much noise, reliable and enough dataset to train our model. This is why, we could train our model effectively and achieve such good performance. We also tested our model on 97 covid images and 157 normal images randomly collected from Kaggle out of which the model correctly classified 92 images as covid i.e. approximately 94% our model predicts correctly whether the x-ray is of a positive patient or a person who has tested negative to the virus as can be observed below. Further we integrated our model into a web app developed using Streamlit and deployed it on Heroku application exposing it to the url: https://covid-19-prediction-system.herokuapp.com screens of which are shared below (Fig. 8-10).

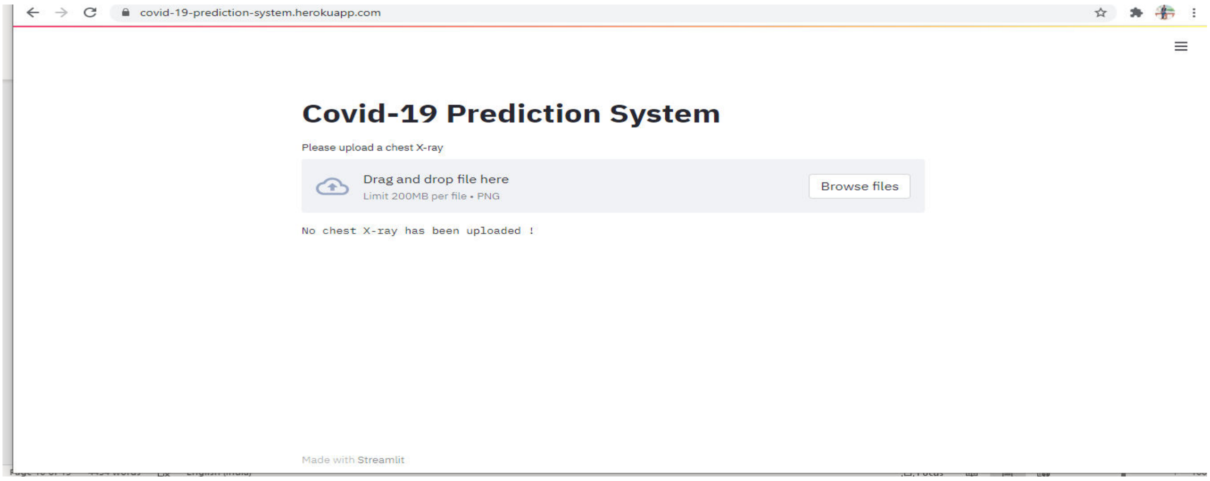


Fig. 8. On first load of the web application

Fig. 9. Result when a chest x-ray of covid positive patient is uploaded

European Journal of Molecular & Clinical Medicine

ISSN 2515-8260 Volume 08, Issue 03, 2021

Our web app can be accessed through laptop or desktop or even in mobile phones. Thus, it provides a platform through which you can easily get your result predicted within seconds by just uploading your chest radiographs.

Fig. 10. Result when a chest x-ray negative for covid is uploaded

Though most of the times, model predicts correctly but it is always advisable to consider this as a second opinion or as a guiding tool for doctors but the result should always be correlated clinically with your regular physician and then only proceed with the appropriate treatment. This model is highly useful in areas where there is shortage of RT-PCR kits and with the increasing number of cases each day, this automated model will help avoid long queues in labs thus aiding in controlling the spread of infection.

6 Future Work

Right now since the model is deployed on cloud, it can be used only if you have a strong internet connection which can be a bottleneck for those who cannot afford internet connection or a smart phone. Therefore as a future work, alternative approach can be thought of by which it can be made to predict even when offline without the need for a stable network. Other future aspect can be to integrate the proposed model with X-ray machine itself in order to get real time predictions within seconds instead of the huge time taken in preparing conventional X-ray report. Also, such automated models can be integrated into walking scanners placed at airports, railway station and malls which will give us real time result as the person pass through it and these results can be monitored through a mobile app in order to store the details of the persons scanned and in case they report to be positive, can be put in isolation as soon as possible and further treatment can be decided upon. This way the chain of transmission among humans can be broken.

References

1. Mall P.K., Singh P.K. and Yadav D., "GLCM Based Feature Extraction and Medical X-RAY Image Classification using Machine Learning Techniques," 2019 IEEE Conference on Information and Communication Technology, 2019, pp. 1-6, doi: 10.1109/CICT48419.2019.9066263.
2. Dutta P., Roy T. and Anjum N., "COVID-19 Detection using Transfer Learning with Convolutional Neural Network," 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2021, pp. 429-432, doi: 10.1109/ICREST51555.2021.9331029.
3. Deep Deb S. and Kumar Jha R., "COVID-19 detection from chest X-Ray images using ensemble of CNN models," 2020 International Conference on Power, Instrumentation, Control and Computing (PICC), 2020, pp. 1-5, doi: 10.1109/PICC51425.2020.9362499.
4. Carvalho E.D., De Carvalho Filho A.O., De Sousa A.D. and De Andrade Lira Rabúlo R., "COVID-19 diagnosis in CT images using CNN to extract features and multiple classifiers," 2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE), 2020, pp. 425-431, doi: 10.1109/BIBE50027.2020.00075.
5. Sethi R., Mehrotra M. and Sethi D., "Deep Learning based Diagnosis Recommendation for COVID-19 using Chest X-Rays Images," 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 1-4, doi: 10.1109/ICIRCA48905.2020.9183278.
6. Degadwala S., Vyas D. and Dave H., "Classification of COVID-19 cases using Fine-Tune Convolution Neural Network (FT-CNN)," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 2021, pp. 609-613, doi: 10.1109/ICAIS50930.2021.9395864.
7. Tahir H., Iftikhar A. and Mumraiz M., "Forecasting COVID-19 via Registration Slips of Patients using ResNet-101 and Performance Analysis and Comparison of Prediction for COVID-19 using Faster R-CNN, Mask R-CNN, and ResNet-50," 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), 2021, pp. 1-6, doi: 10.1109/ICAECT49130.2021.9392487.
8. Islam M.R. and Matin A., "Detection of COVID 19 from CT Image by The Novel LeNet-5 CNN Architecture," 2020 23rd International Conference on Computer and Information Technology (ICCIT), 2020, pp. 1-5, doi: 10.1109/ICCIT51783.2020.9392723.
9. Dong S. and Zhang Z., "Joint Optimization of CycleGAN and CNN Classifier for COVID-19 Detection and Biomarker Localization," 2020 IEEE International Conference on Progress in Informatics and Computing (PIC), 2020, pp. 112-118, doi: 10.1109/PIC50277.2020.9350813.