ID3801 Open Ended Lab/Project

FINAL REPORT

on

Detection of Covid-19 virus using Deep Learning from X-rays

Done By :

Mentor : Dr.Chandra Shekar Lakshminarayanan

Students : Suchitra Yechuri (111801043)

Irene Casmir (111801017)

Acknowledgement

We extend our sincere gratitude to our mentor Dr. Chandra Shekar Lakshminarayanan for his valuable support, guidance and suggestions that helped us complete this project.

We would also like to thank everyone who helped us directly or indirectly in completing this project.

Suchitra Yechuri   111801043
Irene Casmir   111801017

TABLE OF CONTENTS

1. Problem Statement

Develop a deep learning model that can effectively and accurately detect Covid-19 from chest X-ray images, by classifying whether the X-ray is of a Healthy/Normal, Pneumonia affected or Covid-19 affected individual.

2. Context of the problem statement

The entire world has been and is still, affected by Covid-19 virus since the first case was reported in December 2019. The virus continues to pose danger all throughout the globe, with the official record of positive cases so far being 20 Crores and the death toll being 50 Lakhs. The virus continues to spread through new variants at a very alarming rate. Most often, an individual does not show any symptoms even if affected with Covid-19 or any of its variants, especially is the case with the latest variants. This leads to the individual not getting the necessary treatment at the right time, which probably leads to difficulties in health conditions or even death, depending upon the patient’s medical history. Another issue with symptomless and undiagnosed Covid-19 cases is that the affected individual may get in contact with many others, spreading the disease to a larger group of people. Hence, detecting the onset of Covid-19 from early stages proves to be very helpful in controlling the spread and providing timely treatment to the affected patients.

It takes nearly two weeks for an affected individual to show the symptoms of Covid-19. So early detection is not so easy. It is noteworthy to mention that chest X-rays and chest CTs have shown more sensitivity in detecting the onset of Covid-19 even from early stages. So the project proposes a deep learning model to use chest X-rays in detecting Covid-19 virus.

3. Relevance of the problem statement

As mentioned in the context of the challenge statement, it is important to detect Covid-19 as early as possible. Today, the most widely used tests for Covid-19 detection are RT-PCR and antigen tests. These tests are performed when a person shows symptoms similar to Covid-19, or either for getting permission to travel or when the person comes under the primary contact of an affected individual. Even then, studies show that a good percentage of the population would have already been affected and would have self-healed from Covid-19, without them knowing that they have contracted the virus. Also, due to the inconvenience caused while performing the tests using nasal tubes and due to the non-uniformity of test rates in the Government and Private sector, people often abstain from taking these tests. This leads to them becoming carriers of the virus whereby they spread the disease to others, in whom the virus may manifest seriously, such that the source cannot be traced back to them.

So solely depending on RT-PCR and Antigen tests may not be the optimal solution. If there’s a way to detect the virus, through normal and common diagnosing methods which is both affordable and comfortable, timely treatment for the affected as well as spread containment can be ensured. This is where the relevance of the problem statement arises.

X-ray is the most common and the most affordable medical imaging technique used for diagnosis. It is more convenient and affordable than other medical imaging techniques and, depending on the place, even more affordable than RT-PCR and Antigen tests. As has been mentioned earlier, radiographic patterns on chest X-rays and CT scans are more sensitive to Covid-19 virus than normal tests. So this project attempts to develop a neural network model based on deep learning that detects Covid-19 from chest X-rays. The model is not fool-proof as is the case with existing tests for Covid-19, but the model makes predictions with reasonably good accuracy. Hence, depending on the confidence of the prediction made by the model on an X-ray, asymptomatic Covid-19 cases can be detected and isolated at the right time.

4. Adopted Solution

4.1 Dataset

The dataset used for training and testing the model consists of 2957 chest X-ray images, collected and compiled from different open sources. A total of 902 Healthy,1440 Pneumonia and 615 Covid-19 chest X-ray images have been obtained from the following sources :

- 280 images from the GitHub repository[1]
- 60 images from the github repository[2]
- 1288 images from kaggle repository[3]
- 1329 images from kaggle repository[4]

4.2 Approach and Method

The approach used for solving the problem statement was to first preprocess all the images in the dataset. The size of the dataset was increased by choosing 200 preprocessed, random images from each of the three classes and augmenting them. The dataset was split into training, validation and test sets. A neural network model was developed using Transfer learning, to be trained with the dataset and be further used for making predictions from a given X-ray image. Three pre-trained models were experimented to understand the best one with the highest performance for the evaluation metrics. Out of the three models considered - Resnet50, InceptionV3 and Xception - the model using Xception in the transfer learning part gave the highest performance. A simple web interface was developed using Flask whereby users can upload a chest X-ray image and get the prediction made by the model, along with the confidence with which the prediction is made.

The methods used for problem solving are detailed below :

[1] https://github.com/ieee8023/covid-chestxray-dataset
[2] https://github.com/agchung/Figure1-COVID-chestxray-dataset
[3] https://www.kaggle.com/prashant268/chest-xray-covid19-pneumonia
[4] https://www.kaggle.com/tawsifurrahman/covid19-radiography-database

4.2.1 Preprocessing of images

Since the dataset was compiled using images from different sources, images had to be preprocessed before being used for training. All the images were resized to (224,224,3) and then normalized with mean = [0.485, 0.456, 0.406] and standard deviation = [0.229, 0.224, 0.225].The *transforms.Compose* from *torchvision* package was used for preprocessing the images.

4.2.2 Augmentation of images

The dataset has a total of 2957 images.To increase the size of the dataset for a better performing model, augmentation was performed on 200 randomly chosen images from each of the three classes. Hence 600 augmented images were obtained. Augmentation was implemented using the *torchvision* package from pytorch. Using *transforms* class from torchvision, all the 600 images chosen for augmentation,

- were subjected to random rotation with rotation range of 15°
- were subjected to GaussianBlur with kernel size 5 and sigma = (0.1,2.0)

The augmented images were added along with the preprocessed images to finally compile a dataset with 3557 images.

4.2.3 Assigning labels and dataset splitting

The three classes of images were assigned respectively with the labels :

Normal - 0, Pneumonia - 1, Covid - 2

Using *sklearn.model_selection*, out of a total of 3557 images in the dataset, 2845 images were chosen for training, 356 images for validation and 356 images for testing.

4.2.4 Model

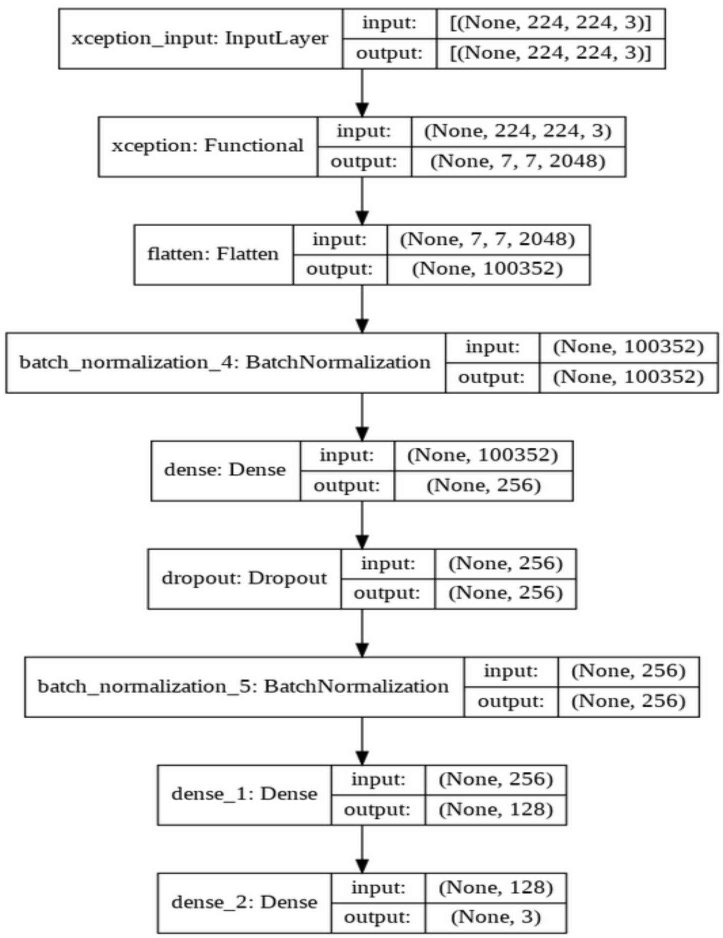
The model used for training makes use of Transfer Learning. Transfer learning is the process where an already trained CNN network with saved weights is reused as the starting point for a model on another task. Transfer learning provides an advantage because the initial layers of the network are already trained and also the network has learned basic features like recognising edges and shape of the image etc. This increases the performance of the network without needing a lot of data and also saves the training time. The last layers of the network were formed according to the number of classes in which the dataset has to be distinguished.

The Xception model was used for the transfer learning process, with all the layers freezed except the last block to save time and computational cost.

The network parameter values used in training were :

- number of layers: 78
- number of epochs: 20
- optimizer: adam
- loss function: sparse categorical cross entropy
- activation function of the last layer: softmax

Three different pre-trained models were trained and tested on the dataset and the best performance among the three was given by Xception. Hence, the Xception model was chosen for the Transfer Learning part. A comparison among the three models tested is given in section 6.

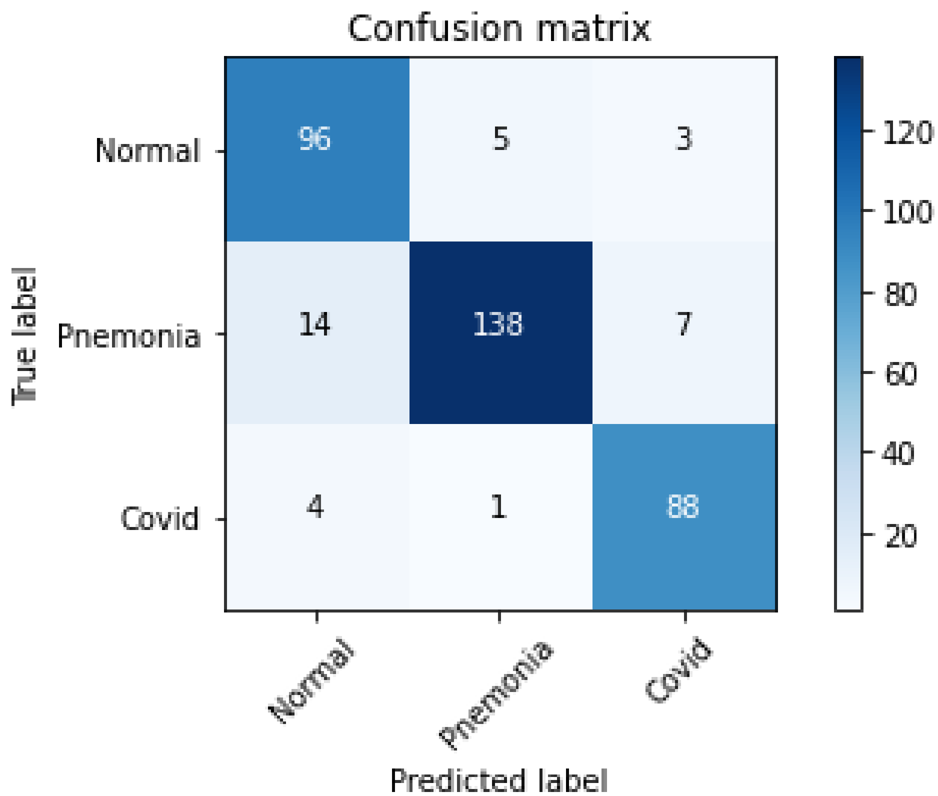


Architecture of the model used

5. Evaluation metrics of the model

5.1 Confusion matrix

The confusion matrix is a table that represents the prediction results of a classification problem. In the table the index (i,j) identifies the number of observations that are of the ith class and are predicted to be in the jth class.



5.2 Accuracy

Accuracy of a model denotes the ratio between the total number of correct predictions to the total number of predictions made by the model. It denotes how well a model performs across all classes under prediction.

```
Accuracy = (True Positives + True Negatives) / Total no. of samples
```

For the Xception model the accuracy is **0.9044**.

5.3 Recall

Recall is defined as the ratio of total number of true positives in the prediction to the sum of true positives and false negatives.

```
Recall = True Positives / (True Positives + False Negatives)
```

The recall values for the Normal, Pneumonia and Covid samples are as follows:
[0.92307692 0.86792453 0.94623656]

5.4 Precision

Precision is the ratio between the number of positive samples correctly classified to the total number of samples classified as positive i.e. ratio between true positives to the sum of true positives and false positives.

```
Precision = True Positives / (True Positives + False Positives)
```

The precision values for the Normal, Pneumonia and Covid samples are as follows:
[0.84210526 0.95833333 0.89795918]

5.5 F1-score

This metric takes into account both the precision and the recall scores. It is the harmonic mean of these scores. It determines the model's accuracy on a dataset.

```
F1-score = 2 x (precision x recall) / precision + recall
```

The F1-scores for the Normal, Pneumonia and Covid samples are as follows:
[0.88073394 0.91089109 0.92146597]

5.6 Specificity

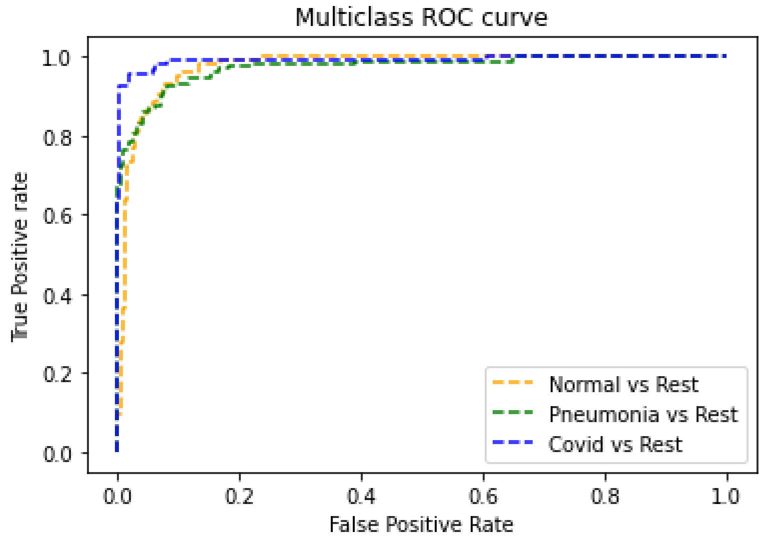
This measure is also known as the True Negative Rate. It tells us about what fraction of the negative samples are correctly classified as negative.

```
Specificity = True Negatives / (True Negatives + False Positives)
```

The specificity values for the Normal, Pneumonia and Covid samples are as follows:
[0.92857143 0.96954315 0.96197719]

5.7 ROC curve

This curve gives the trade-off between the True Positive Rate (TPR) and the False Positive Rate (FPR). This metric gives us a measure of how well our model is able to distinguish between the classes.



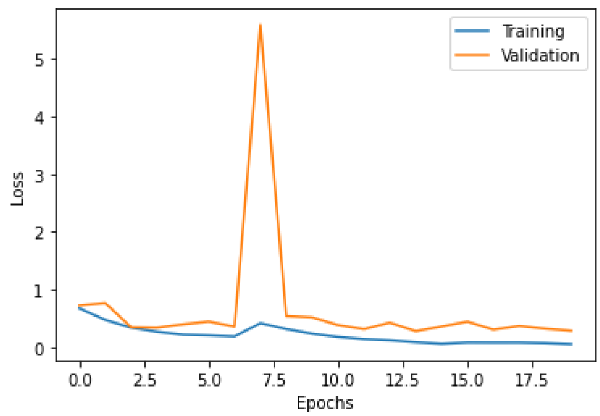
6. Comparison between the models

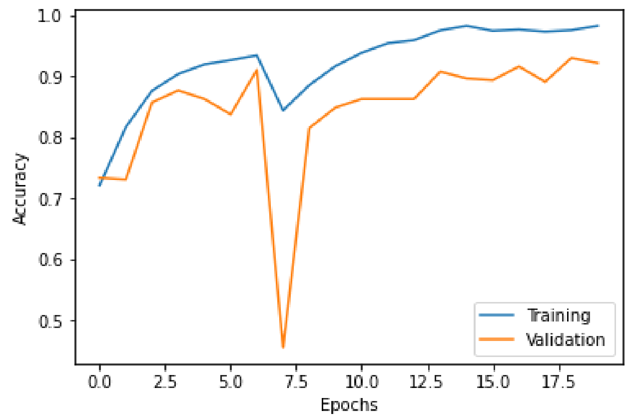
The accuracy given by Xception, Resnet50 and InceptionV3 respectively are 90%, 84% and 77%.

	Xception			**Resnet50**			**InceptionV3**		
	Normal	Pneumonia	Covid	Normal	Pneumonia	Covid	Normal	Pneumonia	Covid
Recall	0.92	0.87	0.95	0.92	0.83	0.97	0.89	0.82	0.56
Precision	0.84	0.96	0.90	0.84	0.94	0.88	0.77	0.71	0.98
Specificity	0.93	0.97	0.96	0.93	0.96	0.95	0.88	0.74	0.99
F1 Score	0.88	0.91	0.92	0.88	0.88	0.92	0.83	0.76	0.71

7. Results

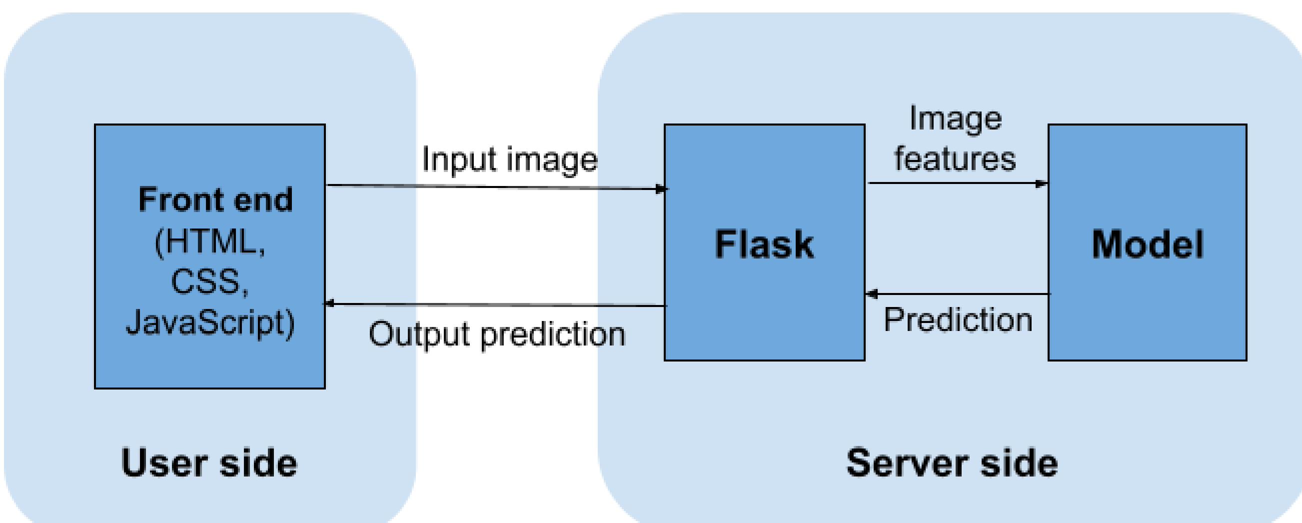
The model implemented using Xception at the start and custom layers at the final part gave a training accuracy of 98% ,validation accuracy of 92% and test set accuracy of 90%. The performance plot for the model is given below:





8. Application

The model was then deployed using the Flask framework. This framework is easy to use and lightweight, which is written in python. HTML, CSS and JavaScript were used to create the template for the application.



8.1 Running the application

1. **Clone the github repository**

You can either download the content or just clone the repository.

2. **Setting up the python environment**

A virtual python environment helps us by keeping the dependencies required by this project isolated from the dependencies of other sources. You can create a virtualenv and activate it, with the following commands:

```
virtualenv -p python3 venv
source venv/bin/activate
```

3. **Installing the requirements**

We can install the requirements for this project with the following command:

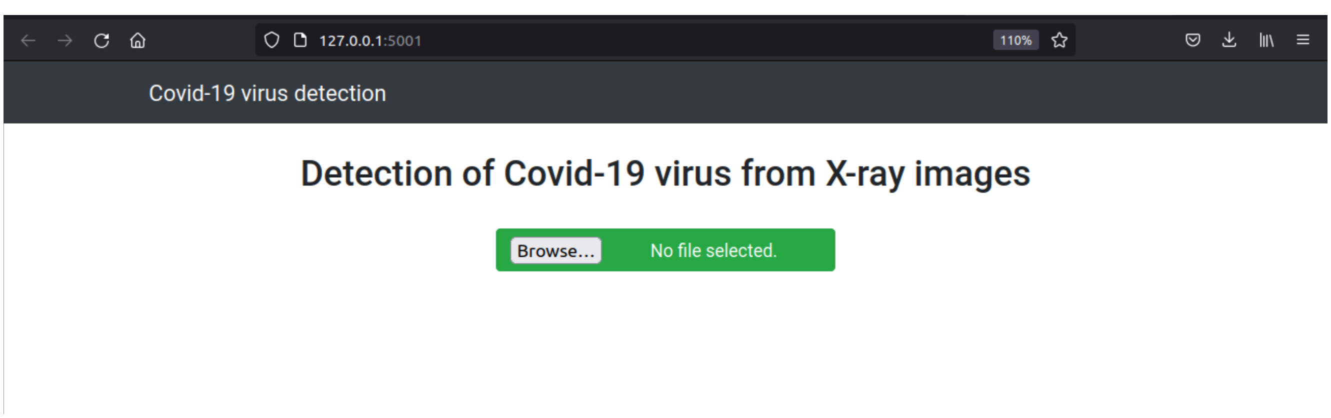
```
pip install -r requirements.txt
```

4. **Finally run the application with**

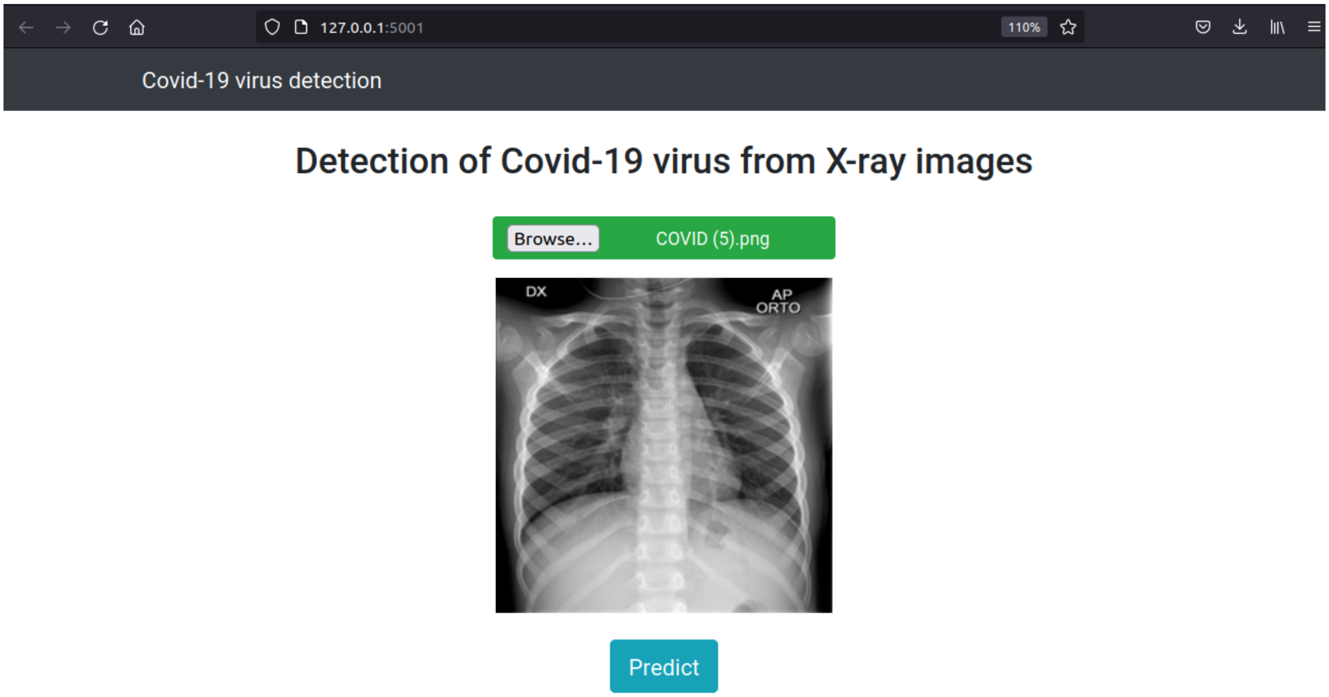
```
python3 app.py
```

8.2 Demo

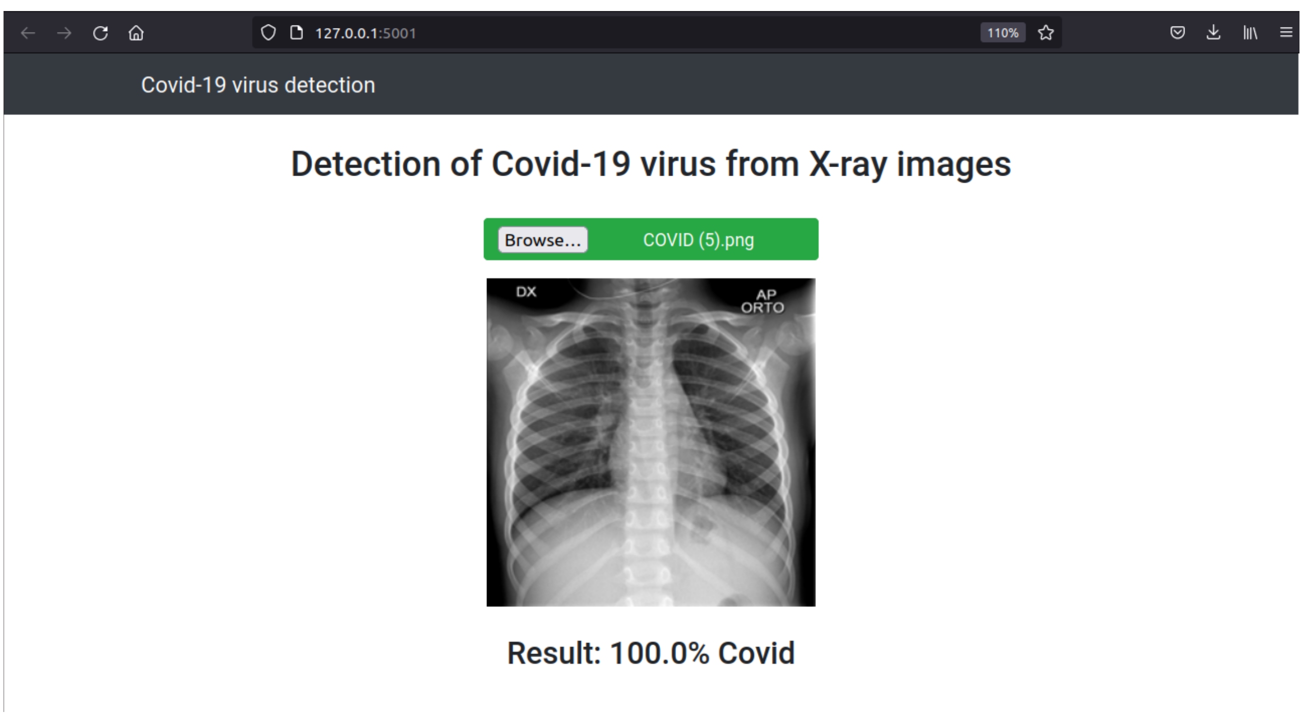
This is the screenshot of the application which allows us to browse an image that we want to predict. The image format can be either jpg, png or jpeg.

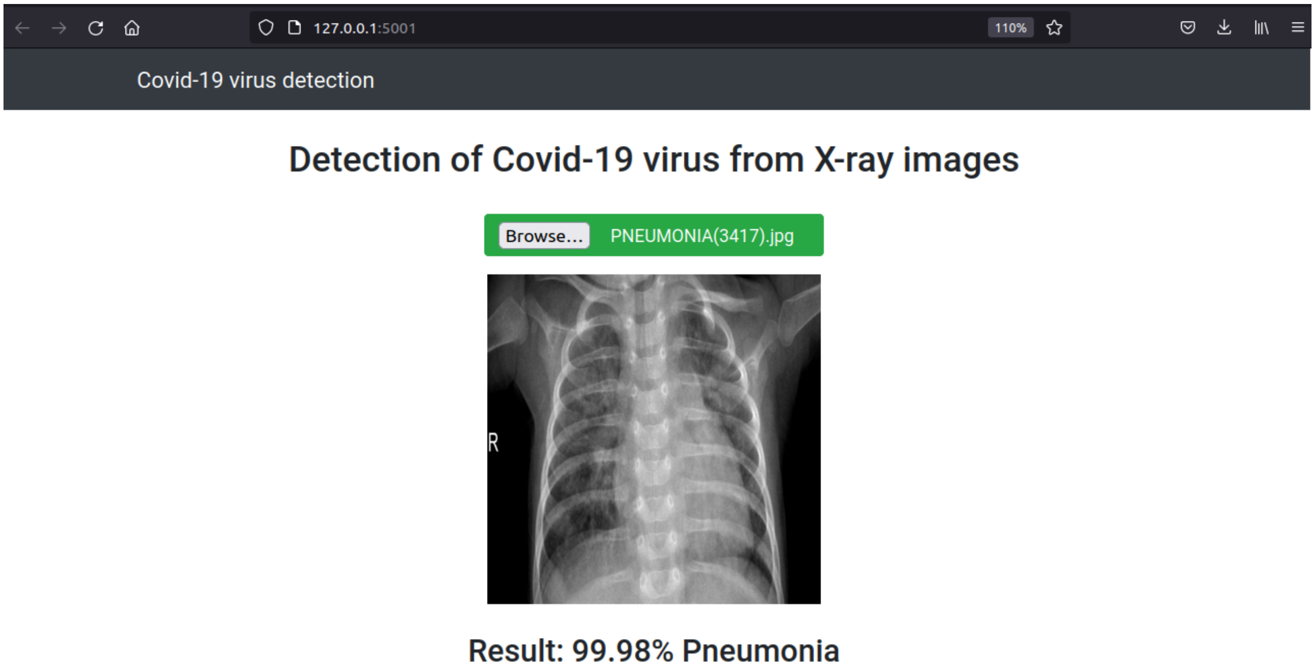


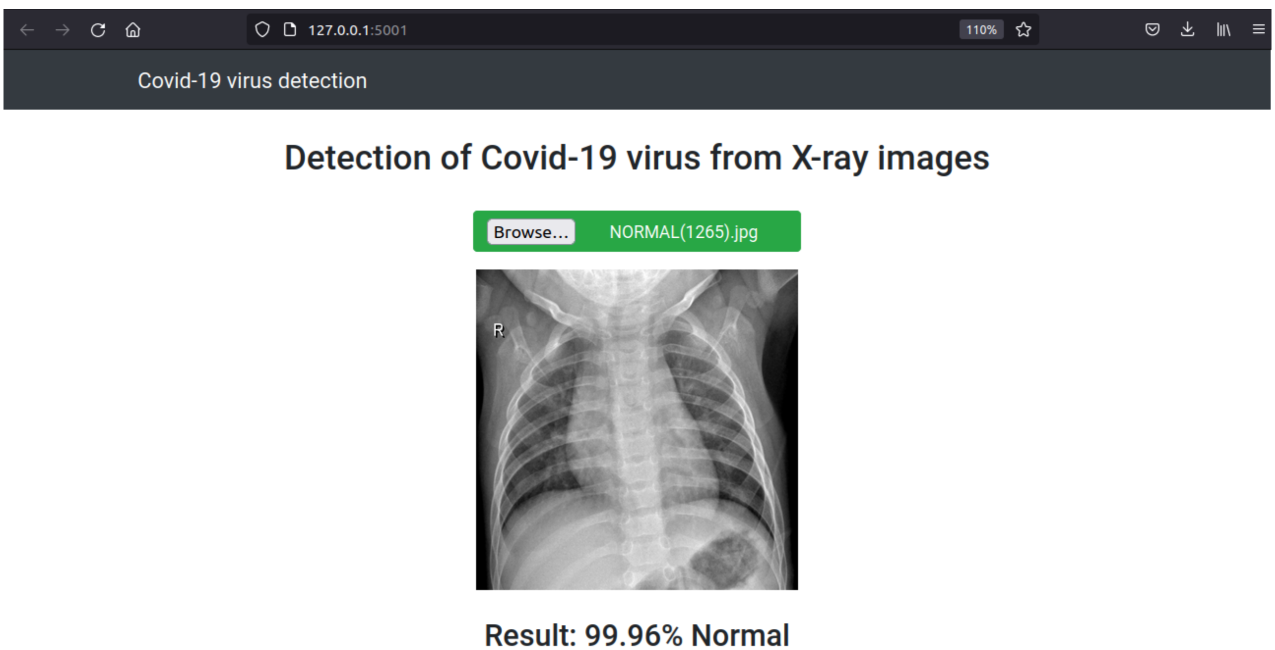
After we give the input image the predict button is displayed.



A few examples of the working of the application are shown below:




127.0.0.1:5001
110%
Covid-19 virus detection
Detection of Covid-19 virus from X-ray images
Browse...
PNEUMONIA(3417).jpg
R
Result: 99.98% Pneumonia


127.0.0.1:5001
110%
Covid-19 virus detection
Detection of Covid-19 virus from X-ray images
Browse...
NORMAL(1265).jpg
R
Result: 99.96% Normal

10. Conclusion

With the increasing threat posed to humanity by Covid-19 and its variants, the need for increasing Covid tests and timely detection of positive cases is of great importance. Through this project, we have made an attempt to explore the feasibility of Deep Learning methods with regard to detecting Covid-19 from chest X-ray images. We have experimented with various CNN models and the best performance was given by Xception. This project is not a perfect solution for contactless testing of Covid-19, but it can be improved and its feasibility for practical medical diagnosis has to be verified by experts.

In future, we would like to enhance the model's performance by training with a larger dataset and by making possible changes to the network, so as to improve the classification accuracy.

11. Contribution

Irene Casmir : Took the lead on preparing the dataset compiled from different sources for training, via preprocessing and augmentation. Developed and experimented with the models, using Resnet50 and InceptionV3 in the transfer learning part and tuned the hyperparameters of these networks to achieve their best performance.These models were then compared against the Xception model to finally choose the one with highest performance.

Suchitra Yechuri : Explored various model architectures employing transfer learning with the Xception model. Tuned the hyperparameters in order to increase the accuracy. Evaluated and compared the performance of all the models using statistical measures like accuracy, precision, recall, specificity, F1-score and ROC curve. Developed the web application using the flask framework for the user interface.

12. References

1. Jain, R., Gupta, M., Taneja, S. et al. Deep learning based detection and analysis of COVID-19 on chest X-ray images. Appl Intell (2020).https://doi.org/10.1007/s10489-020-01902-1
2. Jain G, Mittal D, Thakur D, Mittal MK. A deep learning approach to detect Covid-19 coronavirus with X-Ray images. Biocybern Biomed Eng. (2020).https://doi.org/10.1016/j.bbe.2020.08.008

13. Appendix

Code:
https://colab.research.google.com/drive/1bq2YtckvUqu7mBjxWx2So3w_aJIlB8fJ?usp=sharing
Repository:
https://gitlab.com/irene_casmir/covid-detection-from-chest-x-rays.git