Covid-19 detection with X-Ray images

Problem statement:

COVID-19 infection began in December 2019 in Wuhan, China has widely spread all over the world. We want to do an experimental project on Covid-19 image classification and test the hypothesis of automated detection of covid-19 using X-ray images.

With enough annotated images, deep learning approaches have demonstrated their superiority over the classifying images. CNN architecture is one of the most popular deep learning approaches with superior achievements in the medical imaging domain.

Diagnosis of COVID-19 is typically associated with the symptoms of pneumonia, which can be revealed by X-ray images. We intend to use CNN to detect X-ray images with covid-19.

Goal:

Inspired by many current machine learning research works on COVID-19, we study the application of deep learning model CNN to classify chest x-ray images based on COVID-19 infection.

Our goal is to develop AI based approaches to predict and understand the infection. Our work will be an open source project.

We also evaluated pre-trained VGG16, ResNet, DenseNet models compared to the results of our model.

Furthermore, This project is not intended for clinical use, and it is only an experiment to see the capability of CNN for classifying Covid-19 images.

Dataset :

Finding Covid-19 X-ray images itself is a challenging task. While there exist large public annotated datasets of chest X-rays, there is no collection of COVID-19 chest X-rays designed to be used for computational analysis.

- **covid-chestxray-dataset**

This the only good quality annotated dataset we found which was also referenced in multiple published papers.

https://github.com/ieee8023/covid-chestxray-dataset

- **Kaggle CoronaHack -Chest X-Ray-Dataset Dataset**

Kaggle Dataset Collection Chest X Ray of Healthy vs Pneumonia (Corona) affected patients infected patients along with a few other categories such as SARS (Severe Acute Respiratory Syndrome) ,Streptococcus & ARDS (Acute Respiratory Distress Syndrome).

This dataset was collected by a team from University of Montreal & 80% dataset collected from different sources. But the main source of the Data is from **covid-chestxray-dataset.**

- **COVIDx CT-2 Dataset**

COVIDx CT-2 Dataset consists of CT-SCAN data of 2837 patients with 1958 covid positive. This is a huge collection of Datasets with 29 GB of CT-SCAN data.

https://www.kaggle.com/hgunraj/covidxct

- **COVID-XRay-5K DATASET**

We will this dataset as our main training set, This dataset contains around 5000 images, https://github.com/shervinmin/DeepCovid

This Dataset is extracted from several publications and also with the help of a board-certified radiologist only X-rays with clear signs of COVID-19 are kept.

Dataset preprocessing :

We have 84 covid-19 positive X-ray images which is not enough to train a model. That's why we augmented the dataset. After augmentation we were able to get 242 new augmented images.

The Augmentation code is given below:

```
ImageDataGenerator(rotation_range =5,
```

```
                              rescale=1./255,
                              shear_range=0.2,
                              zoom_range=0.3,
                              horizontal_flip = True,
                              fill_mode = 'nearest',
                              data_format='channels_last',
                              brightness_range=[0.2,1.0])
```

Model Description :

- **Our Model :**

```
Model: "sequential_43"
_____
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_90 (Conv2D)           (None, 85, 85, 64)        1792

_____
max_pooling2d_80 (MaxPooling (None, 42, 42, 64)        0

_____
dropout_52 (Dropout)         (None, 42, 42, 64)        0

_____
conv2d_91 (Conv2D)           (None, 14, 14, 32)        18464

_____
max_pooling2d_81 (MaxPooling (None, 7, 7, 32)          0

_____
dropout_53 (Dropout)         (None, 7, 7, 32)          0

_____
flatten_33 (Flatten)         (None, 1568)              0

_____
dense_68 (Dense)             (None, 256)               401664

_____
dense_69 (Dense)             (None, 1)                 257
=================================================================
Total params: 422,177
Trainable params: 422,177
Non-trainable params: 0

_____
```

The accuracy of our model was 0.97 and loss 0.06. The deep learning model answers nearly the 0.97 of the chest x-ray images of the patients correctly and distinguishes between the infected and not infected lungs which is the first target region of the human body.

- **Using pre-build model on our dataset**

TensorFlow provides a number of pre-trained models which you can use like VGG16, ResNet, DenseNet, etc.

These models have calculated weights trained on the ImageNet dataset.

However, the top layers of these models can be removed, and we can add layers that are necessary for our specific problems.

That way we can utilize lower levels of these models, because on these levels networks detect features like straight line or curve, and at the same time change higher levels to detect specific features.

We loaded and compiled three architectures: MobileNet, ResNet and DenseNet. Note that for each of these architectures we have set the trainable parameter to False, which means that base models are not trained during the training process, but only the additional layers we have added.

Result and evaluation :

- **Our Model Performance**

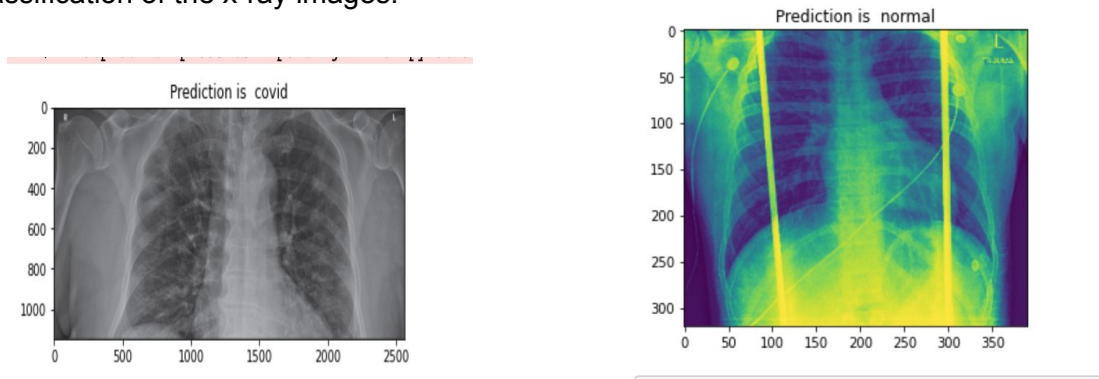For 100 epochs the figure of accuracy and loss are given below :
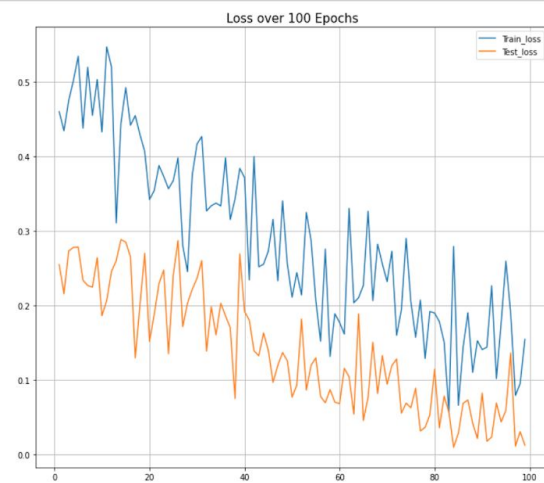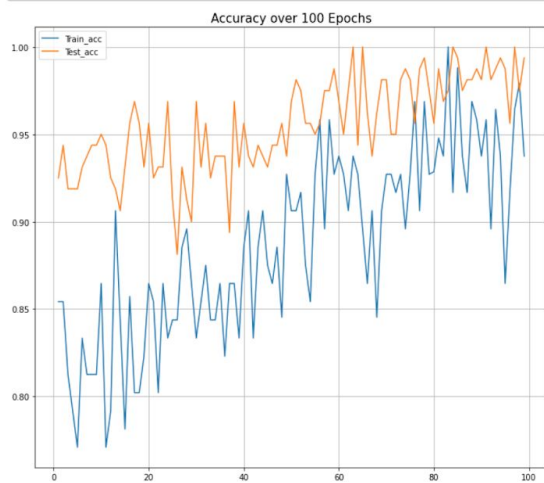

```
--------Our Model---------
Loss: 0.0574
Accuracy: 0.9896
--------------------------
```
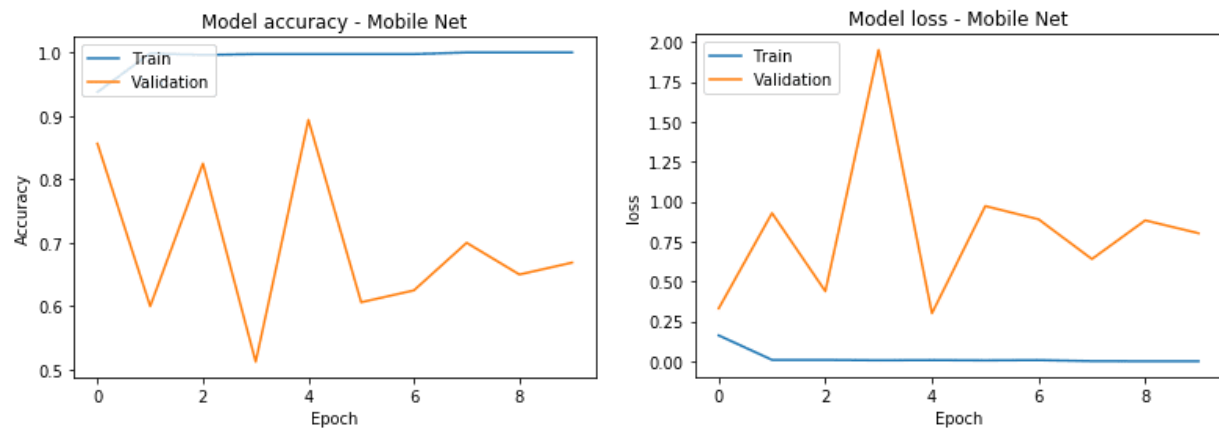
The figure below shows the correct classification of the x-ray images.

Prediction is covid

Prediction is normal

- **Pre- Trained model Performance :**

Here are the plotted results of the training process of the MobileNet:
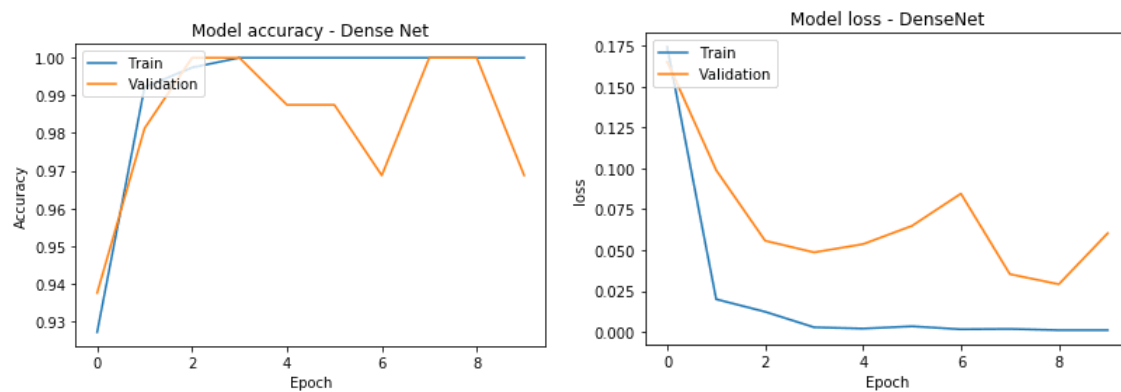


```
--------MobileNet---------
Loss: 0.74
Accuracy: 0.69
--------------------------
```

Here are the plotted results of the training process of the ResNet:
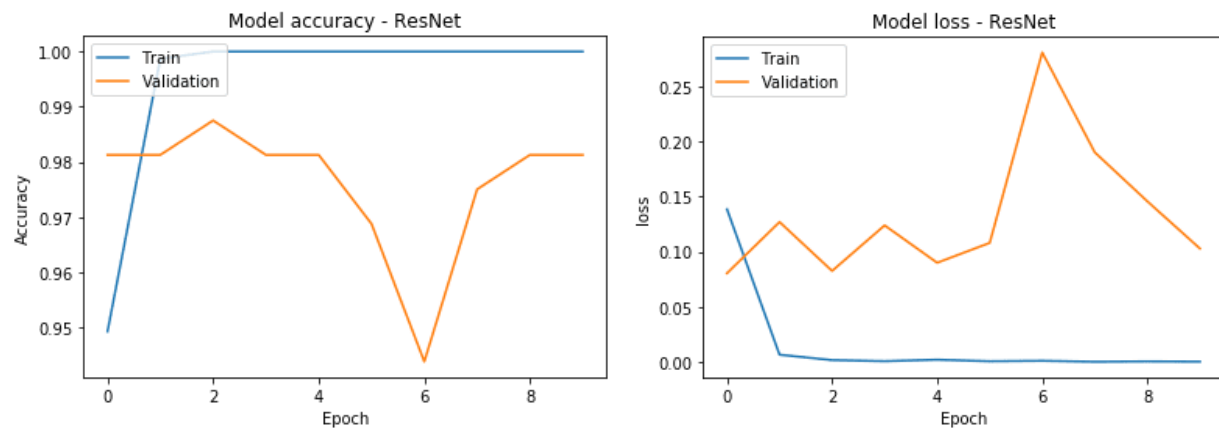

```
---------ResNet------------
Loss: 0.11
Accuracy: 0.98
--------------------------
```

Here are the plotted results of the training process of the DenseNet:


```
---------DenseNet----------
Loss: 0.07
Accuracy: 0.96
--------------------------
```

Future work and Conclusion :

With the availability of large-scale annotated image datasets Paramount progress has been made in deep CNNs for medical image classification. However, the lack of annotated data remains the biggest challenge for detecting COVID-19 infected medical images from non-infected images. Furthermore, Good quality and annotated CT-Scan images can be a better alternative for the X-ray images.

A point to be noted that with a limited number of the medical dataset, it is difficult for us to estimate how our model will perform in the real world.

Our project information and code are available on Github for anyone to work and improve upon it.

Relevant Literature and References:

Implementation Idea:
1. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7372265/
2. https://github.com/shervinmin/DeepCovid

Data set:
1. https://github.com/ieee8023/covid-chestxray-dataset
2. https://github.com/shervinmin/DeepCovid
3. https://www.kaggle.com/praveengovi/coronahack-chest-xraydataset/tasks (kaggle)
4. https://www.kaggle.com/hgunraj/covidxct [CT-Scan]
5. https://github.com/UCSD-AI4H/COVID-CT

Paper or Document:
1. https://www.sciencedirect.com/science/article/abs/pii/S1361841520301584(paper)
2. https://www.pyimagesearch.com/2020/03/16/detecting-covid-19-in-x-ray-images-with-keras-tensorflow-and-deep-learning/
3. https://link.springer.com/article/10.1007/s10489-020-01829-7#Sec15 (paper)
4.
5. https://www.hindawi.com/journals/ijbi/2020/8889023/#data-availability
6. https://arxiv.org/abs/2005.11856 (Predicting COVID-19 Pneumonia Severity on Chest X-ray with Deep Learning)

Notebook related to this (Kaggle):
1. https://www.kaggle.com/timstefaniak/multi-classification-of-x-ray-images
2. https://www.kaggle.com/sidharthavs/covid-19-detection-from-lung-x-rays
3. https://www.kaggle.com/oceancode/corona-image-recognition-analysis
4. https://www.kaggle.com/rahulvv/image-classification-using-efficientnetb7