SRM University AP, Amaravathi

CSE 314 - Digital Image Processing

Project Report

COVID - 19 Diagnosis by Classifying Chest - X-Ray Images using Convolutional Neural Networks

By:- MTH Junaidi, AP17110010074, CSE B

Abstract and Introduction

The coronavirus pandemic, popularly being referred to as the COVID - 19 has put the world under a threatening situation. More than half a million people are suffering and more than a quarter of a hundred thousand people have already lost their lives to this contagion. Statistically, India has still not been affected much yet but necessary precautions are required to prevent the spread of this contagious disease.

In a time of crisis, reliable information and accurate details are required to monitor a country efficiently. However, it is hard to find updated specifics. The current number of cases that have been registered in India differ in various sources. Which of these numbers are true? Which of these can be trusted? Do we have the ability to make the situation under our control with these indefinite facts? Probably not! Surprising, isn't it? Doctors and medical experts believe that a large number of positive cases are actually not yet registered due to the symptoms of the disease being obvious after about 1-14 days. Also, India is able to diagnose only a limited number of samples per day due to the lack of advanced medical technology hence delaying the results. The rate at which this disease can spread is unimaginable. This time period could cost us a spiralling increase in the number of new cases that would have developed due to even a single infected individual.

Keeping the urgency of the situation in mind, I am trying to present a faster, cheaper and a hacky method for the diagnosis of the COVID - 19 disease based on the images of an individual's chest X rays. The idea is to focus on the abnormalities at the lung level to classify whether or not a person is affected by this disease or not. **Also, This project is for the sake of research interest and scope of the project. This should no way be used for medical diagnosis whatsoever unless improved and certified.**

Data and Data Engineering

The data of all the infected chest X Rays of COVID - 19 positive individuals has been obtained from The IEEE Github Repository and the non-COVID - 19 Images has been obtained from the Pneumonia Dataset from Kaggle. These are the images which are both normal and infected with Pneumonia. Both the Images are used for test and train datasets. Here is the Directory tree of the entire data collected, scraped etc.

Method Followed

What we are building is a Machine Learning classification model to classify images of chest X-rays from different angles into COVID19 positive or COVID19 negative. The idea has been drawn since the COVID19 is a respiratory disease. This is a faster method to be able to recognise a slight abnormality in the lung so that the process of diagnosis of the disease would speed up. We will be building a conventional Convolutional Neural Network (CNN) and train it on around 300 images and validate it on around 100 Images.

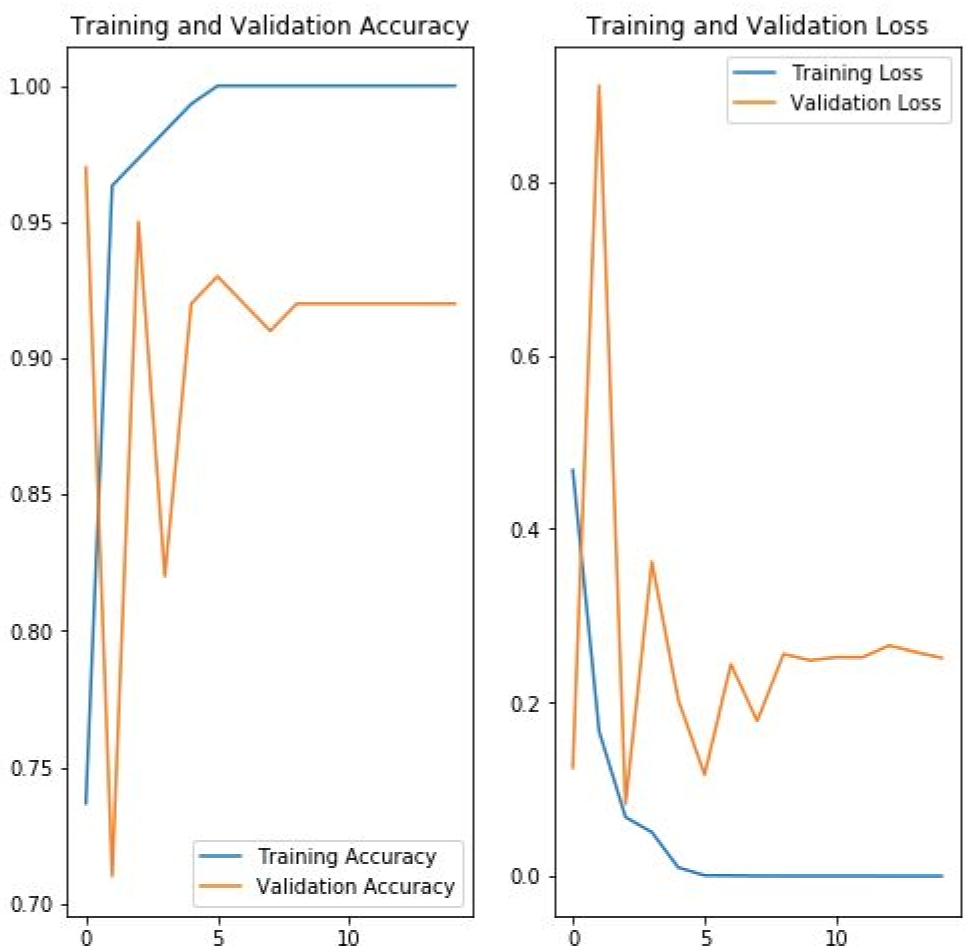
300 and 100 are arbitrary numbers chosen in a range to optimise training time while maintaining accuracy at a comparatively low specs machine (MacBook Air 2017 Model)

These are the steps followed:

- We defined paths and structures to various train, test, COVID and non-COVID datasets. We declared variables with the paths which will be used later to load the image data into the memory for training the model batch-wise.
- We defined Image data generators from tf.keras to load the train and test data from the data sets. We used the declared path variables to point to the respective directory
- We now define the model architecture of our Convolutional Neural Network. Below is our model architecture

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_6 (Conv2D)            (None, 150, 150, 16)      448
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 75, 75, 16)        0
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 75, 75, 32)        4640
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 37, 37, 32)        0
_________________________________________________________________
conv2d_8 (Conv2D)            (None, 37, 37, 64)        18496
_________________________________________________________________
max_pooling2d_8 (MaxPooling2 (None, 18, 18, 64)        0
_________________________________________________________________
flatten_2 (Flatten)          (None, 20736)             0
_________________________________________________________________
dense_4 (Dense)              (None, 512)               10617344
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 513
=================================================================
Total params: 10,641,441
Trainable params: 10,641,441
Non-trainable params: 0
```

- We then train this model using our Image data generators which we build in previous steps. We also store all the training data with each epoch in a variable. Data generators flow the data from the dataset directories which are already classified into subdirectories. Refer to the directory tree structure here.
- We then plot the training data to check the accuracy and the decrement in the loss function. We plot this in a graph to visualise properly and understand the stepwise increment in accuracy and decrement in the loss function.
- Here are the plotted graphs from the training data



- We see that the results are good and the model is now ready to use. We can make predictions using this model with fine accuracy. Any new given chest X-Ray image can now be classified.
- We also changed the last layer activation function just to see if we can make the model any better but we noted that the model is already optimised. Very Minor improvements were observed.

Result and Discussion

After training the model, we now have a fairly accurate model with a validation accuracy of 92%. The availability of huge data is less since COVID-19 is a very recent problem. This page gives all the training history.

```
Epoch 1/15
29/30 [============================>.] - ETA: 0s - loss: 0.4820 - ac
c: 0.7310Epoch 1/15
30/30 [==============================] - 25s 847ms/step - loss: 0.46
77 - acc: 0.7367 - val_loss: 0.1251 - val_acc: 0.9700
Epoch 2/15
29/30 [============================>.] - ETA: 0s - loss: 0.1717 - ac
c: 0.9621Epoch 1/15
30/30 [==============================] - 25s 844ms/step - loss: 0.16
66 - acc: 0.9633 - val_loss: 0.9109 - val_acc: 0.7100
Epoch 3/15
29/30 [============================>.] - ETA: 0s - loss: 0.0695 - ac
c: 0.9724Epoch 1/15
30/30 [==============================] - 26s 871ms/step - loss: 0.06
79 - acc: 0.9733 - val_loss: 0.0831 - val_acc: 0.9500
Epoch 4/15
29/30 [============================>.] - ETA: 0s - loss: 0.0520 - ac
c: 0.9828Epoch 1/15
30/30 [==============================] - 20s 652ms/step - loss: 0.05
05 - acc: 0.9833 - val_loss: 0.3622 - val_acc: 0.8200
Epoch 5/15
29/30 [============================>.] - ETA: 0s - loss: 0.0103 - ac
c: 0.9931Epoch 1/15
30/30 [==============================] - 20s 661ms/step - loss: 0.00
99 - acc: 0.9933 - val_loss: 0.2033 - val_acc: 0.9200
Epoch 6/15
29/30 [============================>.] - ETA: 0s - loss: 8.0299e-04
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 22s 723ms/step - loss: 7.77
13e-04 - acc: 1.0000 - val_loss: 0.1168 - val_acc: 0.9300
Epoch 7/15
29/30 [============================>.] - ETA: 0s - loss: 4.1519e-04
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 19s 633ms/step - loss: 5.03
57e-04 - acc: 1.0000 - val_loss: 0.2440 - val_acc: 0.9200
Epoch 8/15
29/30 [============================>.] - ETA: 0s - loss: 1.0772e-04
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 18s 607ms/step - loss: 1.66
48e-04 - acc: 1.0000 - val_loss: 0.1786 - val_acc: 0.9100
Epoch 9/15
29/30 [============================>.] - ETA: 0s - loss: 8.0399e-05
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 18s 591ms/step - loss: 8.23
28e-05 - acc: 1.0000 - val_loss: 0.2559 - val_acc: 0.9200
Epoch 10/15
29/30 [============================>.] - ETA: 0s - loss: 4.4776e-05
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 18s 594ms/step - loss: 4.33
10e-05 - acc: 1.0000 - val_loss: 0.2485 - val_acc: 0.9200
Epoch 11/15
29/30 [============================>.] - ETA: 0s - loss: 3.4833e-05
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 22s 739ms/step - loss: 3.36
73e-05 - acc: 1.0000 - val_loss: 0.2520 - val_acc: 0.9200
Epoch 12/15
29/30 [============================>.] - ETA: 0s - loss: 2.8948e-05
- acc: 1.0000Epoch 1/15
30/30 [==============================] - 28s 931ms/step - loss: 2.80
74e-05 - acc: 1.0000 - val_loss: 0.2520 - val_acc: 0.9200
Epoch 13/15
```

With each epoch, the loss or accuracy keeps improving and clearly, there is no significant sign of overfitting. We can also see the model becoming better with each epoch from the graphs plotted. We also have to consider the fact that this data is not enough to bet on the accuracy and use the model for deployment.

Conclusion

We have accomplished the initial aim of the Project. We have successfully built an image classifier which classifies a given chest X-Ray Image of an individual as to whether or not he/she is suffering from Pneumonia due to COVID-19. We achieved a great accuracy with a test accuracy of 92% and with a loss of the scale 10^{-6} which is pretty fine. This model can be used to predict now.

Links to the other parts of the project:

1. Github Repository of the course
2. Project Repository
3. Code Archive - Jupyter Notebook for the Project Code
4. Tree Structure of the Dataset
5. Presentation for the Project

References

- Cohen, P.J., Dao, L., & Morrison, P. (2020, March 25). *COVID-19 Image Data Collection.* *https://arxiv.org/abs/2003.11597v1*

- Mooney, P. (2018*). Chest X-Ray Images (Pneumonia)* *https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia/data#*

- Ojala, A.F. (2019, Oct 13). *Intro to deep learning with Keras.* https://github.com/ikhlaqsidhu/data-x/blob/master/06d-tools-cnn-cv/intro-keras-tf1.ipynb

- San Joaquin, A. (2020, March 17). *Using Deep Learning to detect Pneumonia caused by NCOV-19 from X-Ray Images.* *https://towardsdatascience.com/using-deep-learning-to-detect-ncov-19-from-x-ray-images-1a89701d1acd*

- Tensorflow. (2020). *Image classification.* *https://www.tensorflow.org/tutorials/images/classification*

- Wang, S., Kang, B., Ma, J., Zeng, X., Xiao, M., Guo, J., Cai, M., Yang, J., Li, Y., Meng, X., & Xu, B. (2020, March 16). *A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19).* *https://doi.org/10.1101/2020.02.14.20023028*

- Xu, X., Jiang, X., Ma, C., Du, P., Li, X., Lv, S., Yu, L., Chen, Y., Su, J., Lang, G., Li, Y., Zhao, H., Xu, K., Ruan, L., & Wu, W. (2019). *Deep Learning System to Screen Coronavirus Disease 2019 Pneumonia.* *https://arxiv.org/pdf/2002.09334.pdf*

- Zu, Z.Y., Jiang, M.D., Xu, P.P., Chen, W., Ni, Q.Q., Lu, G.M., & Zhang, L,J. (2020, Feb).
Coronavirus Disease 2019 (COVID-19): A Perspective from China. https://doi.org/10.1148/radiol.2020200490