

Detection of COVID-19 from Chest X-Ray Images Using Convolutional Neural Networks

- Princy Gautam
B20BB051





Introduction

Deep learning for image recognition applications is capable of learning millions of images and models like ConvNet have produced stable and accurate results. Traditional diagnostic techniques, such as PCR tests, can be time-consuming and resource-intensive. Basically, it is a rapid and reliable alternative for COVID-19 detection.



Motivation

1

Using X-ray images for the automated detection of COVID-19 might be helpful in particular for countries and hospitals that are unable to purchase a laboratory kit for tests or that do not have a CT scanner.

3

The financial costs of the laboratory kits used for diagnosis, especially for developing and underdeveloped countries, are a significant issue when fighting the illness.

2

Considering the time required for diagnosis AI and deep learning initiated to support doctors who aim to treat patients and fight the illness.



Model - Convolutional Neural Network (CNN)

The standard CNN design which -

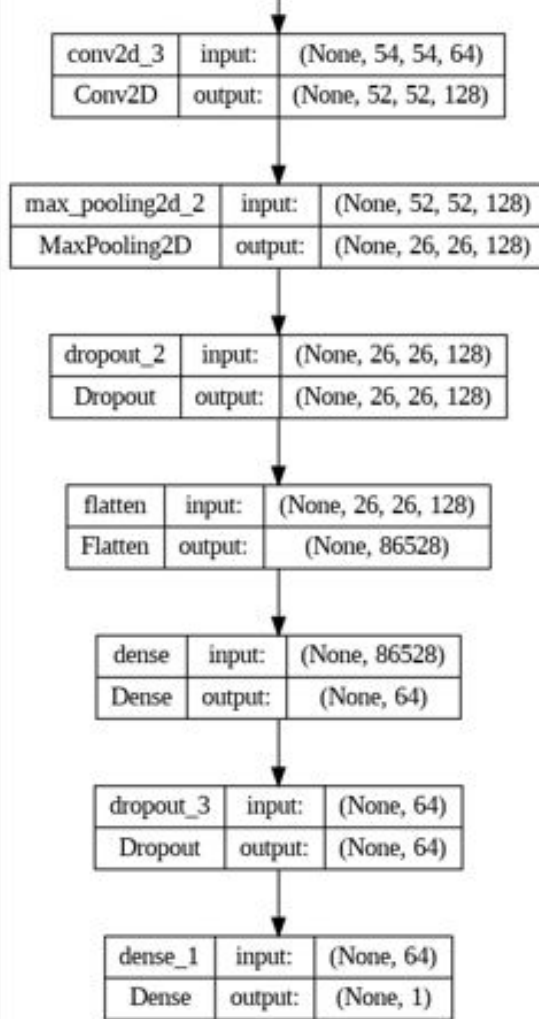
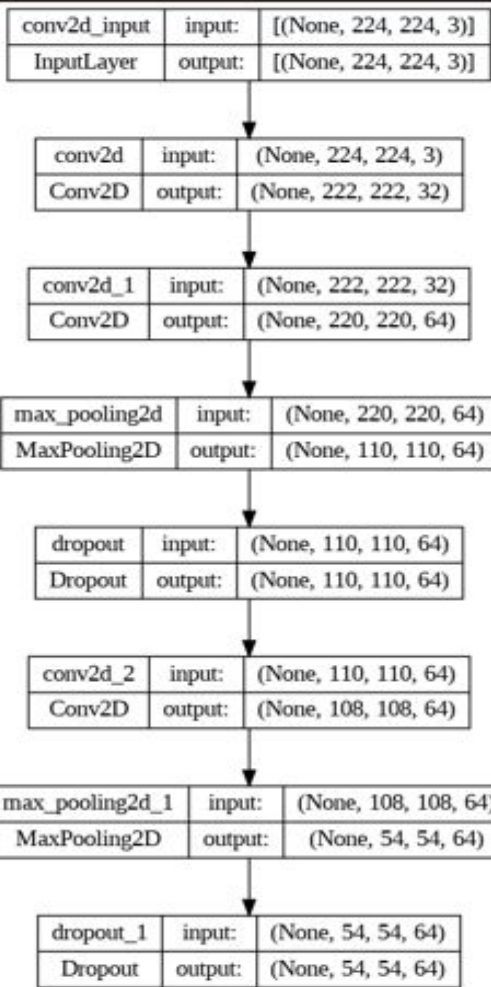
progressively extracts features through
convolutional layers,

pooling layers reducing spatial dimensions
and

dropout for regularization.

The flattened features are then fed into fully
connected layers for binary classification.

Model





Results

The model achieved 98% accuracy on validation set and a training loss of 0.1147 after training for 10 epochs. We achieved less accuracy when training the model with ImgAug library.

```
<ipython-input-10-3b784b638eea>:3: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Pl
hist = model.fit_generator(
Epoch 1/10
7/7 [=====] - 13s 2s/step - loss: 0.8452 - accuracy: 0.5625 - val_loss: 0.6740 - val_accuracy: 0.8667
Epoch 2/10
7/7 [=====] - 11s 2s/step - loss: 0.6086 - accuracy: 0.6562 - val_loss: 0.5618 - val_accuracy: 0.7167
Epoch 3/10
7/7 [=====] - 11s 2s/step - loss: 0.4609 - accuracy: 0.7768 - val_loss: 0.3853 - val_accuracy: 0.9500
Epoch 4/10
7/7 [=====] - 10s 1s/step - loss: 0.2938 - accuracy: 0.8750 - val_loss: 0.1807 - val_accuracy: 0.9500
Epoch 5/10
7/7 [=====] - 11s 1s/step - loss: 0.2393 - accuracy: 0.9107 - val_loss: 0.1505 - val_accuracy: 0.9667
Epoch 6/10
7/7 [=====] - 11s 2s/step - loss: 0.1802 - accuracy: 0.9330 - val_loss: 0.0860 - val_accuracy: 0.9667
Epoch 7/10
7/7 [=====] - 11s 2s/step - loss: 0.3124 - accuracy: 0.8795 - val_loss: 0.1299 - val_accuracy: 0.9667
Epoch 8/10
7/7 [=====] - 11s 2s/step - loss: 0.1722 - accuracy: 0.9330 - val_loss: 0.1063 - val_accuracy: 0.9833
Epoch 9/10
7/7 [=====] - 11s 2s/step - loss: 0.1591 - accuracy: 0.9286 - val_loss: 0.0840 - val_accuracy: 0.9667
Epoch 10/10
7/7 [=====] - 11s 2s/step - loss: 0.1147 - accuracy: 0.9688 - val_loss: 0.0690 - val_accuracy: 0.9833
```



References

<https://journals.sagepub.com/doi/full/10.1177/2472630320958376>

<https://github.com/aleju/imgaug>

<https://www.sciencedirect.com/science/article/pii/S2666827021000694>

<https://www.geeksforgeeks.org/python-data-visualization-using-covid19-india-api/>

<https://www.analyticsvidhya.com/blog/2021/03/binary-cross-entropy-log-loss-for-binary-classification/>

<https://towardsdatascience.com/binary-image-classification-in-pytorch-5adf64f8c781>

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



Thank you.

