31 January, 2021

COVID-19 -VS- VIRAL PNEUMONIA

CONTENTS

Description 1

AlexNet 2

ResNet 3

VGG16 4

Inception Network 5

MobileNet V2 6

U-Net 7

Conclusion 7

References and Links 8

DESCRIPTION

To classify the CXR scans into COVID-19 and Viral Pneumonia, I trained 6 different models on the COVID-19 Radiography Database.

The dataset contained *3616* CXR images for COVID-19 and *1345* CXR images for Viral Pneumonia. This dataset was divided in the ratio, $\frac{Validation\ set\ size}{Total\ dataset\ size} = 0.1$, giving us a total of *4465* images for training and the remaining for validation. For testing all the trained models, I used the Chest X-ray (Covid-19 & Pneumonia) dataset, from which I took a total of *460* images for COVID-19 and *855* for Pneumonia.

Wherever possible, I tried to use transfer learning using weights pretrained on the *ImageNet*. This includes ResNet50, VGG16 and MobileNet V2.

Moreover, to avoid overtraining the model (and save time) I've also implemented *early stopping* to halt the training process if the *validation loss* doesn't go down by *0.0001* for *5* epochs.

The Final results are as follows:

AlexNet

Upon training the metrics for model evaluation were as follows:

i) Training metrics:
   (1) loss: 0.0154
   (2) accuracy: 99.6%
   (3) F1 Score: 0.9949
      (a) Precision: 0.9960
      (b) Recall: 0.9960

ii) Validation Metrics:
   (1) loss: 0.0406
   (2) accuracy: 98.79%
   (3) F1 Score: 0.9846
      (a) Precision: 0.9879
      (b) Recall: 0.9879

iii) Test Metrics:
   (1) loss: 0.0361
   (2) accuracy: 99.39%
   (3) F1 Score: 0.9933
      (a) Precision: 0.9939
      (b) Recall: 0.9939

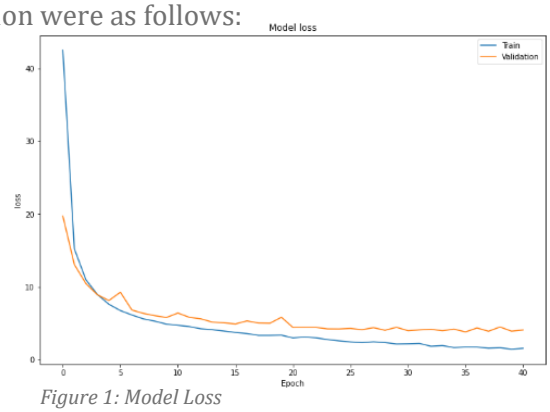


Figure 1: Model Loss

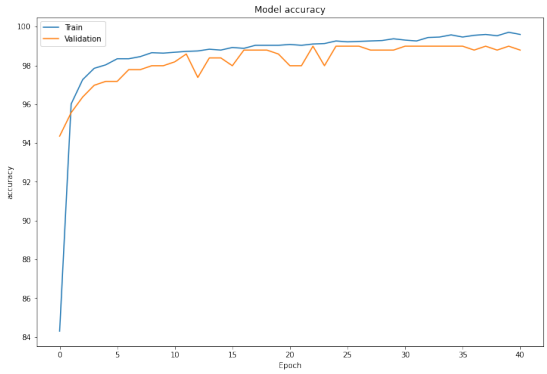
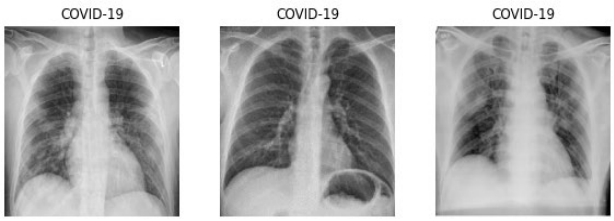
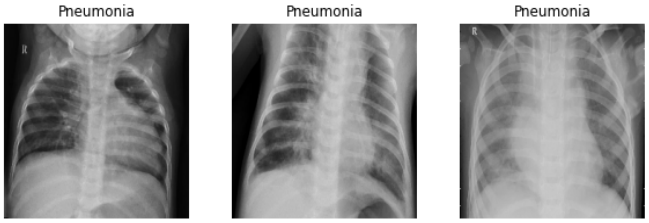


Figure 2: Model Accuracy

Prediction Examples:




```
This image most likely belongs to COVID with a 73.01 percent confidence.
This image most likely belongs to COVID with a 73.11 percent confidence.
This image most likely belongs to COVID with a 73.11 percent confidence.
```




```
This image most likely belongs to VP with a 72.99 percent confidence.
This image most likely belongs to VP with a 73.10 percent confidence.
This image most likely belongs to VP with a 73.11 percent confidence.
```

ResNet

For the ResNet model, I used **ResNet50** as base with weights pretrained on *ImageNet.* The output of the base network is then passed on to the following layers:

```
A0 = base.output

A1 = layer.GlobalAveragePooling2D( )(A0)
N1 = layer.BatchNormalization()(A1)
N1 = layer.Dropout(0.1)(N1)

A2 = layer.Dense(units=3072, activation='relu')(N1)
N2 = layer.BatchNormalization()(A2)
N2 = layer.Dropout(0.2)(N2)

A3 = layer.Dense(units=512, activation='relu')(N2)
N3 = layer.BatchNormalization()(A3)
N3 = layer.Dropout(0.4)(N3)

outputs = layer.Dense(units = 2, activation = 'softmax')(N3)
```

The final metrics of the model after training are:

i) Training metrics:
   (1) loss: 0.6748
   (2) accuracy: 99.71%
   (3) F1 Score: 0.9963
      (a) Precision: 0.9971
      (b) Recall: 0.9971

ii) Validation metrics:
   (1) loss: 0.6755
   (2) accuracy: 99.60%
   (3) F1 Score: 0.9948
      (a) Precision: 0.9960
      (b) Recall: 0.9960

iii) Test metrics:
   (1) loss: 0.6729
   (2) accuracy: 99.62%
   (3) F1 Score: 0.9958
      (a) Precision: 0.9962
      (b) Recall: 0.9962

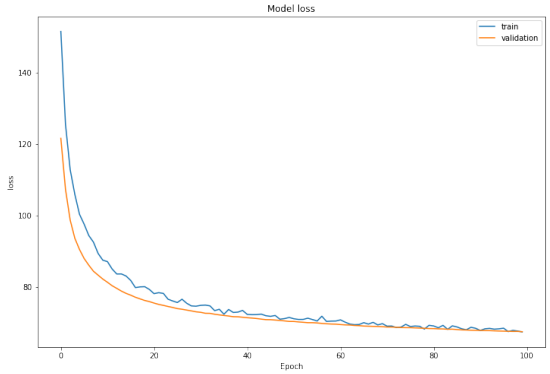


Figure 3: Model Loss -vs- Epochs

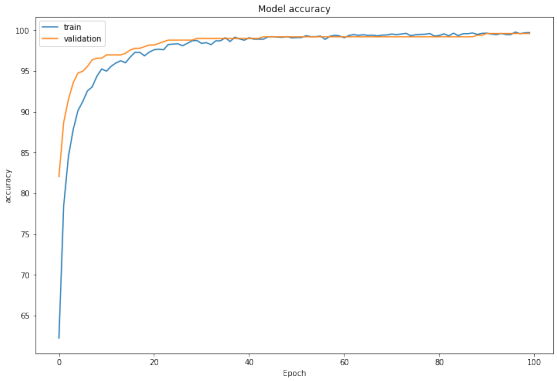


Figure 4: Model Accuracy -vs- Epochs

VGG16

Similar to the ResNet model, this model was also made by passing the output of VGG16 base network to the network architecture described in ResNet section. The base network was pretrained with *ImageNet* weights.

The results after training are:

i) Training metrics:
   (1) loss: 0.304
   (2) accuracy: 95.23%
   (3) F1 Score: 0.9423
      (a) Precision: 0.9523
      (b) Recall: 0.9523

ii) Validation metrics:
   (1) loss: 0.2465
   (2) accuracy: 96.98%
   (3) F1 Score: 0.9626
      (a) Precision: 0.9698
      (b) Recall: 0.9698

iii) Test metrics:
   (1) loss: 0.2259
   (2) accuracy: 97.49%
   (3) F1 Score: 0.9721
      (a) Precision: 0.9749
      (b) Recall: 0.9749

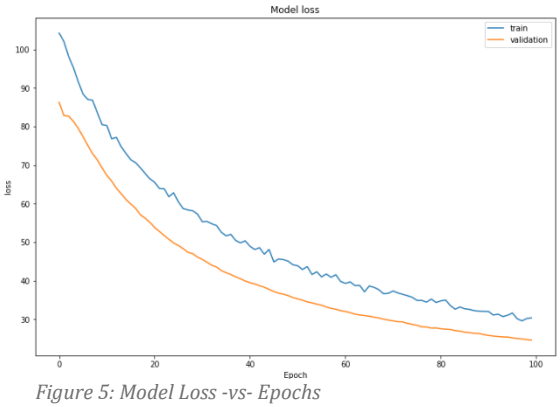


Figure 5: Model Loss -vs- Epochs

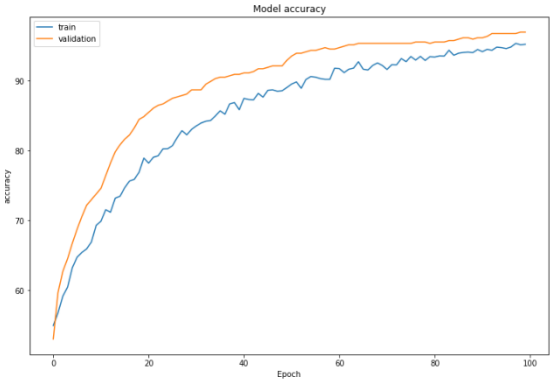
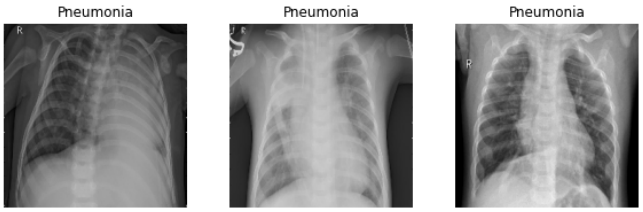


Figure 6: Model Accuracy -vs- Epochs

Prediction Examples:


```
This image most likely belongs to VP with a 60.59 percent confidence.
This image most likely belongs to COVID with a 73.09 percent confidence.
This image most likely belongs to COVID with a 73.04 percent confidence.
```




```
This image most likely belongs to VP with a 73.10 percent confidence.
This image most likely belongs to VP with a 73.04 percent confidence.
This image most likely belongs to VP with a 73.11 percent confidence.
```

Inception Network

For the *Inception Network* model, I used an opensource implementation of *GoogLeNet* and then passed its output to the same architecture described in the ResNet Section.

The results after training are:

i) Training metrics:
   (1) loss: 0.4837
   (2) accuracy: 99.13%
   (3) F1 Score: 0.9889
      (a) Precision: 0.9913
      (b) Recall: 0.9913

ii) Validation metrics:
   (1) loss: 0.5196
   (2) accuracy: 99.19%
   (3) F1 Score: 0.9897
      (a) Precision: 0.9919
      (b) Recall: 0.9919

iii) Test metrics:
   (1) loss: 0.4804
   (2) accuracy: 99.09%
   (3) F1 Score: 0.99
      (a) Precision: 0.9909
      (b) Recall: 0.9909

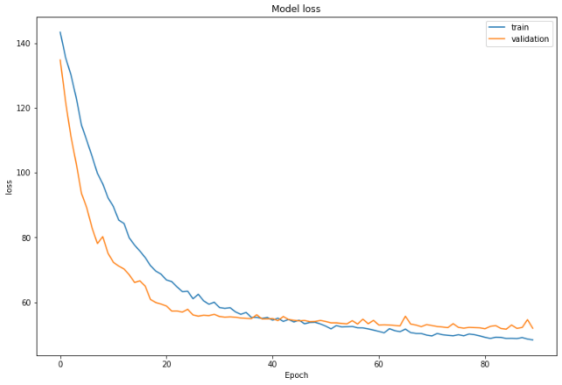


Figure 7: Model Loss -vs- Epochs

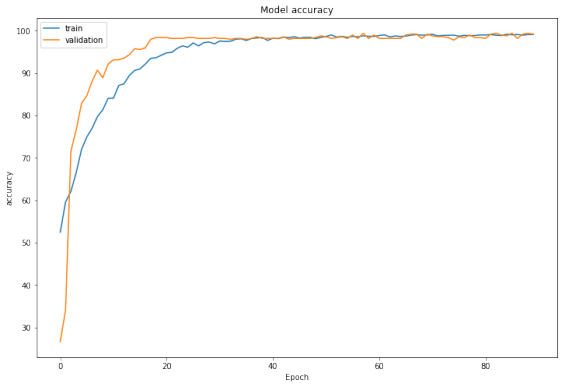
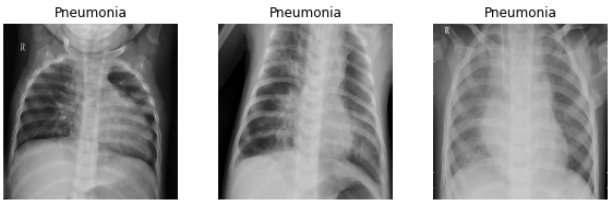


Figure 8: Model Accuracy -vs- Epochs

Prediction Examples:


```
This image most likely belongs to COVID with a 73.09 percent confidence.
This image most likely belongs to COVID with a 73.10 percent confidence.
This image most likely belongs to COVID with a 73.11 percent confidence.
```




```
This image most likely belongs to VP with a 73.11 percent confidence.
This image most likely belongs to VP with a 73.11 percent confidence.
This image most likely belongs to VP with a 73.11 percent confidence.
```

MobileNet V2

Similar to ResNet model, this model uses MobileNet V2 as the base network with weights pretrained on *ImageNet.* The output of the base network is then passed on to the same network architecture described in the ResNet section for the final output.

The results after the training are:

i) Training metrics:
   (1) Loss: 0.3572
   (2) accuracy: 94.91%
   (3) F1 Score: 0.9262
      (a) Precision: 0.9391
      (b) Recall: 0.9424

ii) Validation metrics:
   (1) loss: 0.2849
   (2) accuracy: 97.78%
   (3) F1 Score: 0.9722
      (a) Precision: 0.9778
      (b) Recall: 0.9778

iii) Test metrics:
   (1) loss: 0.7571
   (2) accuracy: 68.29%
   (3) F1 Score: 0.6826
      (a) Precision: 0.6829
      (b) Recall: 0.6829

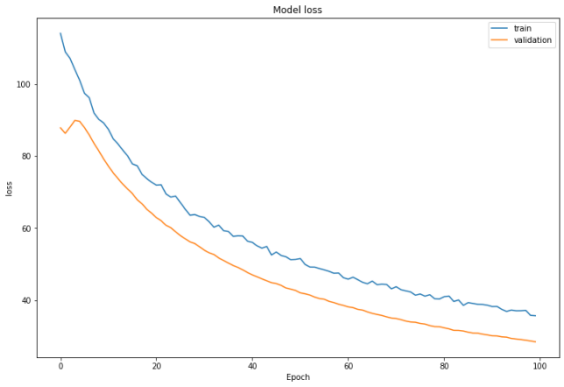


Figure 9: Model Loss -vs- Epochs

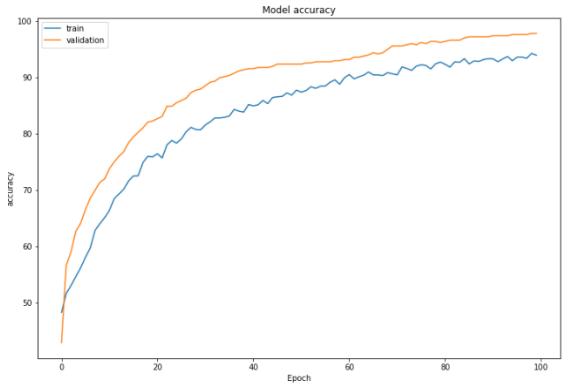


Figure 10: Model Accuracy -vs- Epochs

Prediction Examples:


```
This image most likely belongs to VP with a 54.17 percent confidence.
This image most likely belongs to VP with a 69.79 percent confidence.
This image most likely belongs to COVID with a 66.37 percent confidence.
```

U-Net

This model is implemented using U-Net as the base network, the segmentation output of which is then passed on the network architecture described earlier in the ResNet Section.

The results upon training are as follows:

i) Training metrics:
   (1) loss: 0.2686
   (2) accuracy: 97.67%
   (3) F1 Score: 0.9706
      (a) Precision: 0.9767
      (b) Recall: 0.9767

ii) Validation metrics:
   (1) loss: 0.2235
   (2) accuracy: 99.19%
   (3) F1 Score: 0.9896
      (a) Precision: 0.9919
      (b) Recall: 0.9919

iii) Test metrics:
   (1) loss: 0.3090
   (2) accuracy: 96.12%
   (3) F1 Score: 0.9578
      (a) Precision: 0.9612
      (b) Recall: 0.9612

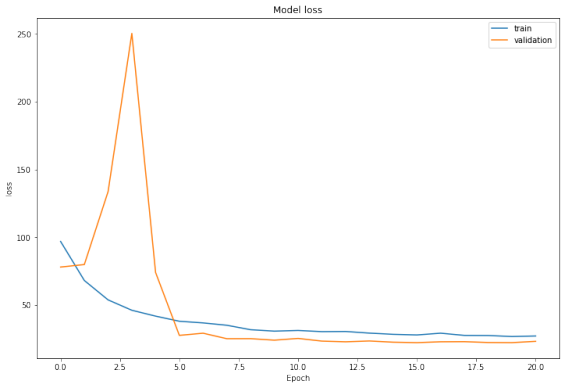


Figure 11: Model Loss -vs- Epochs

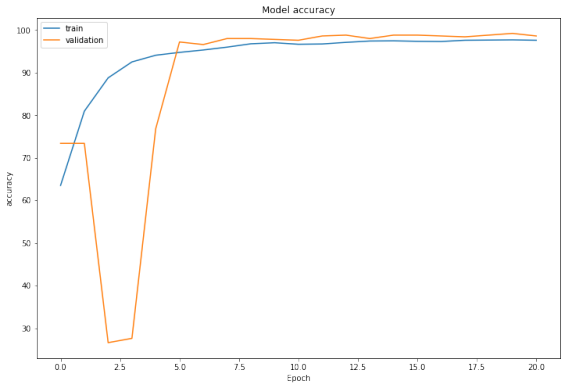


Figure 12: Model Accuracy -vs- Epochs

CONCLUSION

Since *dropout* was used for all the models except AlexNet, the validation metrics are expected to be better than the training metrics for these models. Based on the results we see that the ResNet model performs the best. The model has nearly the same performance on the validation set as the training set and it generalizes really good to previously unseen data (the test set). It has the highest *accuracy* and *F1 score* of all the models on the test set.

InceptionNet is the second-best choice for the classification purpose since it has evaluation metrics similar to that of the ResNet and the difference between the performance on the training set and the validation set is also very small. It also generalized really well to the test set.

REFERENCES AND LINKS

GitHub Repository

- *Minor Project sem-3.ipynb* is the jupyter notebook in which all the models are trained.
- *ModelTesting.ipynb* is the jupyter notebook in which all the models were tested on the test set created from unseen data.

COVID-19 Radiography Database

Chest X-ray (Covid-19 & Pneumonia)

References

- U-Net: Convolutional Networks for Biomedical Image Segmentation