Report On
Detection of Pneumonia from Chest X-Ray Images using ResNet50 and VGG16

Shameen Shrestha

Experimental Design

1. **Data Collection:**
   The dataset of Chest X-ray images was from Kaggle.The dataset includes 5863 images in the training dataset, 16 images in the validation dataset, and 624 images in the test dataset. The images are classified into "Normal" and "Pneumonia" in all the datasets.

2. **Data Preprocessing:**
   In the data preprocessing step, the images are resized and normalized by dividing by 255. In the training dataset, different data augmentation techniques are applied. Data augmentation is useful to improve the performance and outcomes of machine learning models by forming new and different examples to train datasets, this helps improve the predictive accuracy and general performance of machine learning models by reducing the risk of overfitting

3. **Model Selection:**
   Two pre-trained models; ResNet50 and VGG16 were fine-tuned on the training dataset to detect the presence of pneumonia in chest X-ray images.

   I. ResNet50: ResNet50-50 is a convolutional neural network that is 50 layers deep. ResNet50, short for Residual Networks, is a classic neural network used as a backbone for many computer vision tasks.
   II. VGG16:
   VGG16, or VGGNet, is a convolutional neural network (CNN) model that includes 16 layers. It consists of 13 layers of convolutional and pooling layers, followed by three fully connected layers.

4. **Model Fine Tuning:**

   I. In the ResNet50, in the pre-trained model the last layer was removed and replaced with a global pooling average layer followed by a dense layer with

1024 units and ReLU activation. And the final layer is a dense layer with 2 units and a softmax activation function for binary classification. The ResNet50 model is fine-tuned using binary-cross entropy loss and Adam Optimizer with a learning rate of 0.001.

II. In the VGG, the top layer of the pre-trained model is removed then a dense layer with 256 units and ReLU activation is added to learn more complex features, and finally, an output dense layer with 2 units and softmax activation is added to predict the probabilities of the two classes. The model is then compiled using the Adam optimizer with a learning rate of 0.0001 and the categorical cross-entropy loss function.

5. **Evaluation of Models:**
   The models are evaluated on the test set to measure their performance. The evaluation metrics used are accuracy, precision, recall, F1 score, and the AUC-ROC curve. Also, a confusion matrix is generated to visualize the performance of the model on different classes.

Models

I. <u>ResNet50</u>

ResNet50 is a deep convolutional neural network architecture. The name ResNet50 comes from "Residual Network" and "50" represents the number of layers in the network.

The main innovation of ResNet50 is the use of residual connections, which allow the network to be deeper while avoiding the vanishing gradient problem. To tackle the vanishing gradient problem, ResNet50 uses “skip connection”; to skip one or more layers and pass the input directly to a deeper layer in the network. This way, the gradient can flow directly from the deeper layer to the earlier layers, preventing it from vanishing.

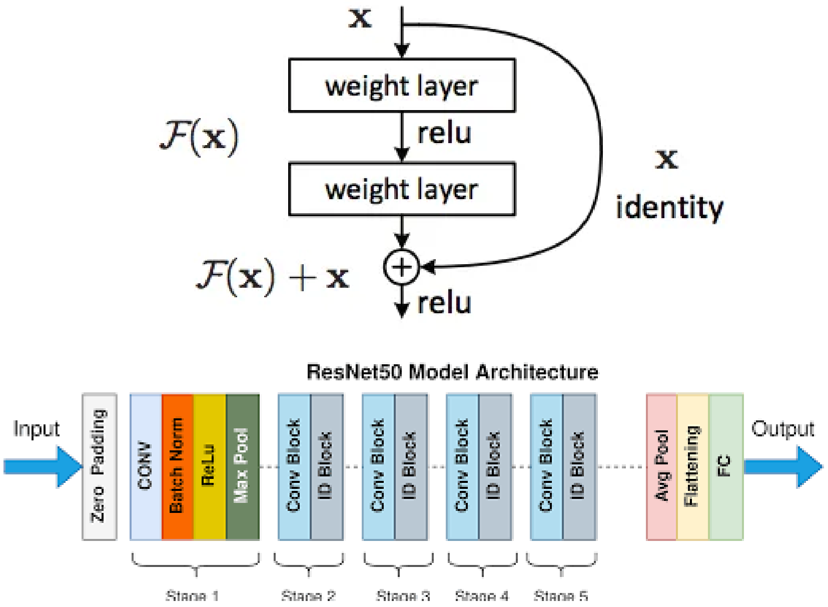


Fig: ResNet50 Architecture

The architecture of ResNet50 consists of several layers of convolutional neural network (CNN) blocks followed by a global average pooling layer and a fully connected layer. Each CNN block consists of several convolutional layers and batch normalization layers, with a shortcut connection added that bypasses one or more blocks to connect the input directly to the output of a later block.

1. **Convolutional layer**: The input image is fed into the network, and the first layer applies a convolution operation with a 7x7 kernel size and a stride of 2, followed by batch normalization and ReLU activation.

2. **ReLu Activation**:It sets all negative values to zero, effectively introducing non-linearity and allowing the network to learn complex representations of data. This helps in avoiding the vanishing gradient problem and also speeds up the training process.

3. **Max-pooling layer**: A max-pooling layer with a 3x3 kernel size and a stride of 2 is applied to reduce the size of the feature map.

4. **Residual blocks:** The core of the ResNet50 architecture is the residual block, which consists of multiple convolutional layers and skip connections. Each residual block has two 3x3 convolutional layers, followed by batch normalization and ReLU activation. The output of the first convolutional layer is added to the input of the residual block, which is then passed through the second convolutional layer. This process allows the network to learn the residual mapping, which helps in training deeper networks.

5. **Global average pooling**: After the residual blocks, a global average pooling layer is applied to convert the feature map into a 1-dimensional vector.Global average pooling is a technique that replaces the traditional fully connected layer in a neural network with a pooling operation. It takes the average of all the activations in each feature map and outputs a single value for each feature map. This reduces the total number of parameters in the model and helps prevent overfitting.

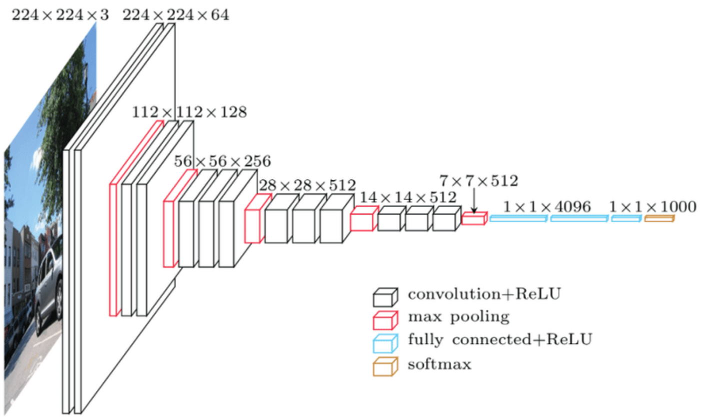
6. **Fully connected layer**: A single fully connected layer with softmax activation is used to output the final classification probabilities. The

number of units in the fully connected layer is equal to the number of classes in the dataset.

In conclusion, ResNet50s is a highly effective type of neural network architecture incorporating skip connections to improve gradient flow, leading to more effective training of deep neural networks. This, in turn, enables better performance in a range of computer vision tasks, including image recognition, object detection, and segmentation.

II. VGG16

VGG16 refers to the VGG model, also called VGGNet. It is a convolution neural network (CNN) model supporting 16 layers. It consists of only 3x3 convolutional layers, which makes it easy to implement and understand. Additionally, the use of small convolutional filters helps to improve performance and reduce the number of parameters in the model. VGG also achieved state-of-the-art performance on image classification tasks during its time, demonstrating the effectiveness of deep convolutional neural networks for computer vision tasks.



A VGG network consists of small convolution filters. VGG16 has three fully connected layers and 13 convolutional layers. The architecture is explained below:

- **Input**: VGGNet receives input of image size 224x224.

- **Convolutional laye**r: The convolutional filters of VGG use the smallest possible receptive field of kernel size 3×3 with a stride of 1. VGG also uses a 1×1 convolution filter as the input's linear transformation.Conv-1 Layer has 64 filters, Conv-2 has 128 filters, Conv-3 has 256 filters, Conv 4 and Conv 5 have 512 filters.

- **ReLu activation**: After each convolutional layer, a Rectified Linear Unit (ReLU) activation function is applied element-wise to the output feature map, which introduces non-linearity into the network and helps it learn more complex featuresReLU is a linear function that provides a matching output for positive inputs and outputs zero for negative inputs.

- **Pooling layers**: In VGG, after a set of convolutional layers, a pooling layer is added. Pooling layers reduce the spatial size of the previous layer by down-sampling the feature maps. VGG16 uses max pooling, which takes the maximum value of each non-overlapping window of size 2x2, thereby reducing the size of the feature maps by half. The pooling layers help in controlling overfitting and improving the computational efficiency of the model.

- **Fully connected layer**s: VGGNet includes three fully connected layers. The first two layers each have 4096 channels, and the third layer has 1000 channels, one for every class. However, for our pneumonia detection task, we replaced the last fully connected layer with a dense layer of 2 neurons, as we have only two classes (normal and pneumonia). The output of the final dense layer is passed through a softmax activation function to obtain the probability distribution of the input image belonging to each class.

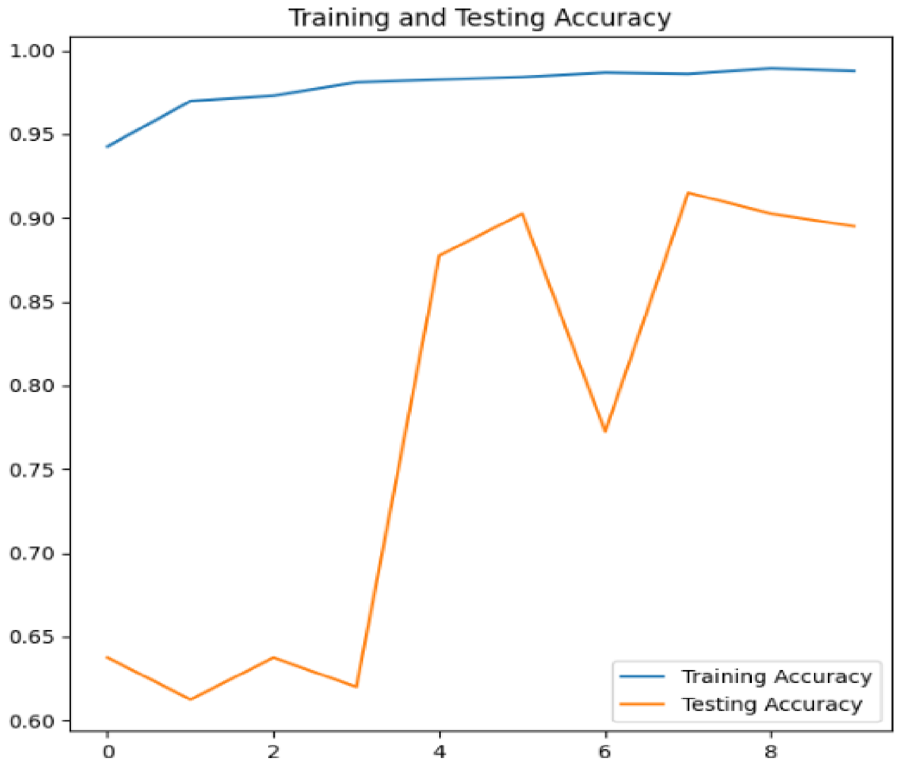
- **Softmax Function**: The softmax function transforms the output of the last layer into a probability distribution over the classes, ensuring that the sum of the probabilities for all classes equals one. This allows us to interpret the output of the models as the probability of an image belonging to each class, which is useful for making predictions and evaluating the performance of the models.

VGG is considered a good architecture for image classification tasks due to its relatively simple and straightforward architecture, making it easy to understand and implement. VGG has a uniform architecture with small convolution filters, which leads to fewer parameters;this property makes VGG more memory-efficient and easier to train.

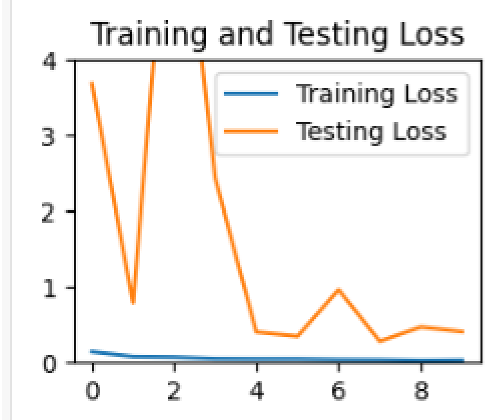
Training Information

I. **ResNet50:**

- The ResNet50 model was trained for 10 epochs with a batch size of 20.The optimizer used was Adam, and the loss function used was binary cross-entropy.
- After being trained for 10 epochs,it achieved an accuracy of 98.77% and a loss of 0.0325. The validation accuracy was 89.50%, with a validation loss of 0.4102.
- The training vs testing accuracy curve shows that the training accuracy increases with every epoch whereas that testing accuracy fluctuates but ultimately increases at the 10th epoch.

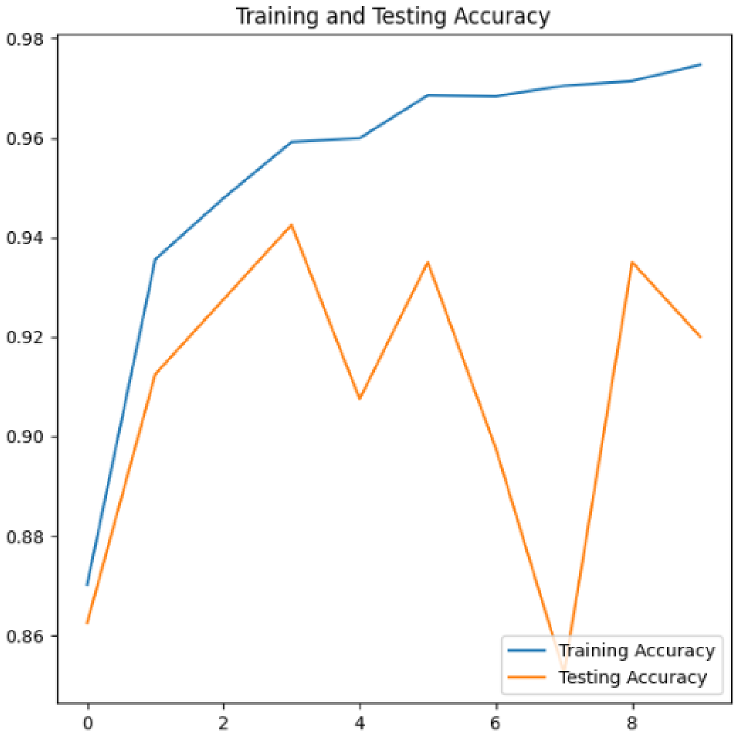


- The training loss vs testing loss graph shows validation loss of 0.4102 is also higher than the training loss of 0.035,the model appears to perform well on the training data, but may need further optimization to improve its performance on the validation data.

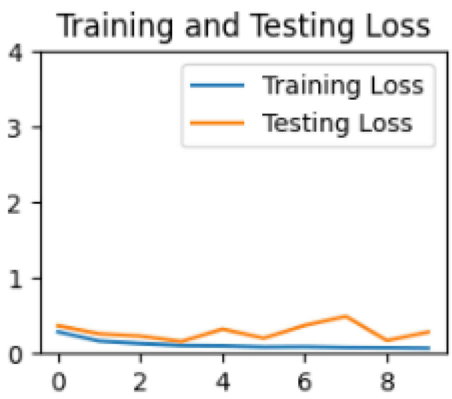


II. VGG:

- The VGG model was trained for 10 epochs with a batch size of 20.The optimizer used was Adam, and the loss function used was binary cross-entropy.
- After being trained for 10 epochs,it achieved training accuracy of 97.47% and a loss of 0.0711. The validation accuracy was 92%, with a validation loss of 0.2841.
- The training vs testing accuracy curve shows that the training accuracy increases with every epoch whereas that testing accuracy fluctuates till the 10th epoch.

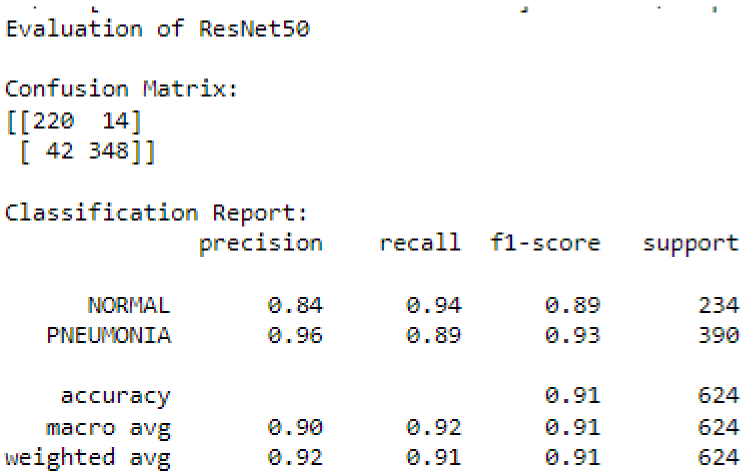


- The training loss vs testing loss graph shows the difference in loss in training and testing . We can see the value of training loss and testing loss is close, which indicates it is not overfitting to the training data and generalizing well in the validation data.



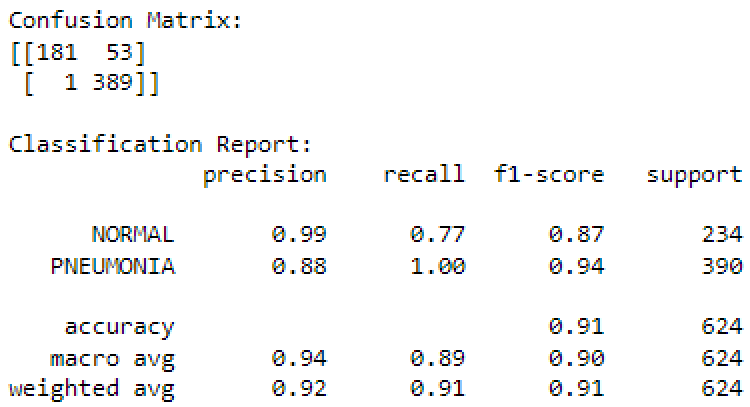
Evaluation Metrics

1. **Confusion Matrix**:A confusion matrix is a table used to evaluate the performance of a classification model. It displays the number of true positives, true negatives, false positives, and false negatives.



```
Evaluation of ResNet50

Confusion Matrix:
[[220  14]
 [ 42 348]]

Classification Report:
              precision    recall  f1-score   support

      NORMAL       0.84      0.94      0.89       234
   PNEUMONIA       0.96      0.89      0.93       390

    accuracy                           0.91       624
   macro avg       0.90      0.92      0.91       624
weighted avg       0.92      0.91      0.91       624
```

a. In the confusion matrix for ResNet50, we can see that. The model predicted 262 samples to be in the "Normal" class, 220 were actually in the "Normal" class while the remaining 42 were misclassified as "Pneumonia". Similarly, out of the 449 samples predicted to be in the "Pneumonia" class, 389 were correctly classified while 60 were misclassified as "Normal".



```
Confusion Matrix:
[[181  53]
 [  1 389]]

Classification Report:
              precision    recall  f1-score   support

      NORMAL       0.99      0.77      0.87       234
   PNEUMONIA       0.88      1.00      0.94       390

    accuracy                           0.91       624
   macro avg       0.94      0.89      0.90       624
weighted avg       0.92      0.91      0.91       624
```

b.Similarly, in the confusion matrix forVGG, we can see that model predicted 234 samples to be in the "Normal" class, out of which 181 samples are actually in the "Normal" class and 53 samples are misclassified and actually belong to the "Pneumonia" class. Similarly, the model predicted 390 samples to be in the "Pneumonia" class, out of which it predicted 389 samples correctly to be in the "Pneumonia" class, but misclassified 1 sample that actually belongs to the "Normal" class.

2. **Accuracy**: Accuracy is the proportion of correctly classified samples out of the total number of samples.
   Accuracy=(TP + TN) / (TP + TN + FP + FN)

In our trained models ResNet50 and VGG16 both achieved accuracy of 91% ; it predicted 568 samples correctly out of a total of 640 samples .We can conclude that in terms of accuracy ResNet50 and VGG16 performed well

3. **Precision:** The proportion of true positive predictions out of the total number of positive predictions. It measures the ability of the model to minimize false positives.
   Precision=TP / (TP + FP)
   a. In ResNet50, the precision achieved for ResNet50 for "Normal" class was 84% and for "Pneumonia" class was 96%. This indicates that ResNet50

was more accurate in identifying "Pneumonia" cases correctly than "Normal" cases.

b. In the VGG16 model, the precision achieved was 99% for “Normal” cases whereas for “Pneumonia” cases it was 88%, which shows that VGG was more accurate in identifying “Normal” cases correctly than “Pneumonia” cases.

4. **Recall:** The proportion of true positive predictions out of the total number of actual positive samples. It measures the ability of the model to identify all positive samples.
   Recall=TP / (TP + FN)
    a. The recall for the “Normal” case in ResNet50 was 94% whereas for “Pneumonia” was 89%. This shows that of all the actual “Normal” cases, the model correctly identified 94% of them as “Normal”. On the other hand, the recall of 89% for “Pneumonia” means that out of all the actual “Pneumonia” cases, the model correctly identified 89% of them as “Pneumonia”.

    b. In VGG16, the recall for “Normal” was 77% and for “Pneumonia” was 100%. This shows that of all the actual “Normal” cases, the model correctly identified 77% of them as “Normal”. On the other hand, out of all the actual “Pneumonia” cases, the model correctly identified all of them as “Pneumonia”.

5. F1 Score: It is the harmonic mean of precision and recall, and is a single metric that balances both metrics.

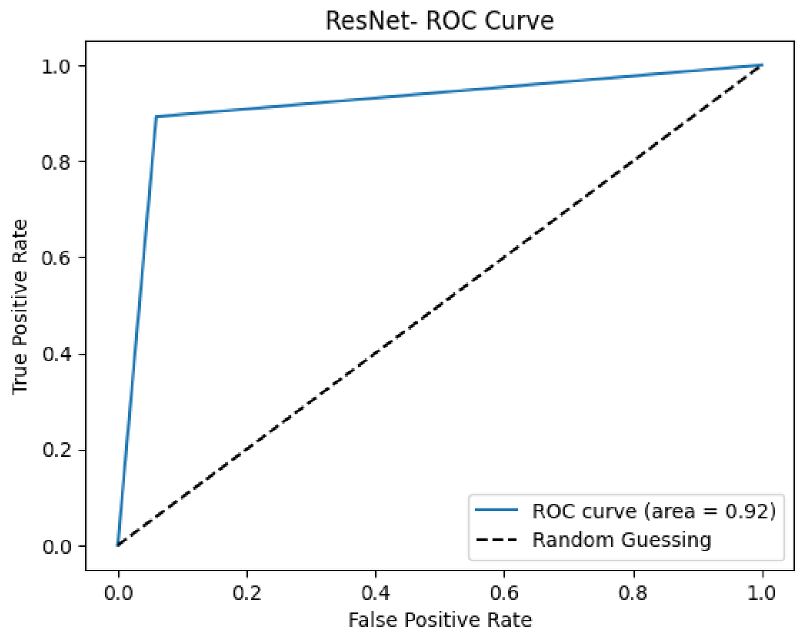
F1 Score=2 * (precision * recall) / (precision + recall)

F1 score is one of the important metrics that takes both precision and recall into account. According to our ResNet50 model's classification report, we can see that the "Normal" case has a higher precision of 84% and a lower recall of 74%,

whereas the "Pneumonia" case has a higher recall of 89% but a comparatively lower precision of 96%.
Similarly, in the VGG model, we can see that “Normal” has higher precision of 99% but a lower 77% recall, and in “Pneumonia” there is a lower precision of 88% but a higher recall of 100%. So, the F1 score is used to balance the trade-off between those two metrics.

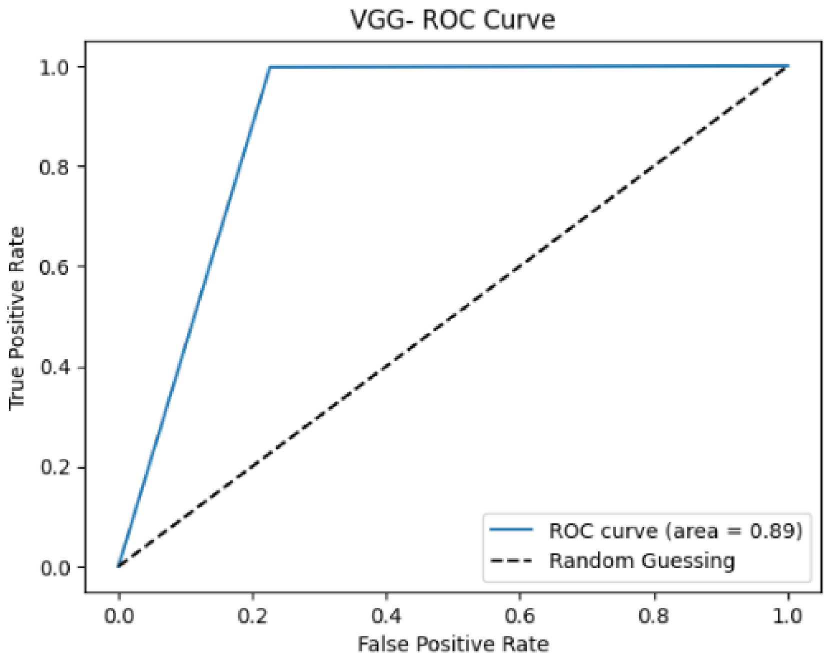
a. In the ResNet50 model, the F1-score for the "Normal" class was 89%, and for the "Pneumonia" class, it was 93%. This indicates that the model performs better at identifying "Pneumonia" cases than "Normal" cases, as evidenced by the higher F1-score for "Pneumonia". The overall F1 score for the model was 91%.
b. In the VGG model, we have an F1-score of 87% for “Normal” and 94% for “Pneumonia”. The F1-score shows that the VGG model performs better at identifying “Pneumonia” cases than “Normal” cases. The overall F1 score for the VGG model would be 90.5%.

6. AUC-ROC: AUC-ROC curve is the graphical evaluation of the performance of the classifier. The curve plots the true positive rate against the false positive rate. The area under the ROC curve (AUC) is also commonly used as a metric to evaluate the performance of a classifier. The AUC ranges between 0 and 1, with higher values indicating better performance.



Roc curve of ResNet50 model

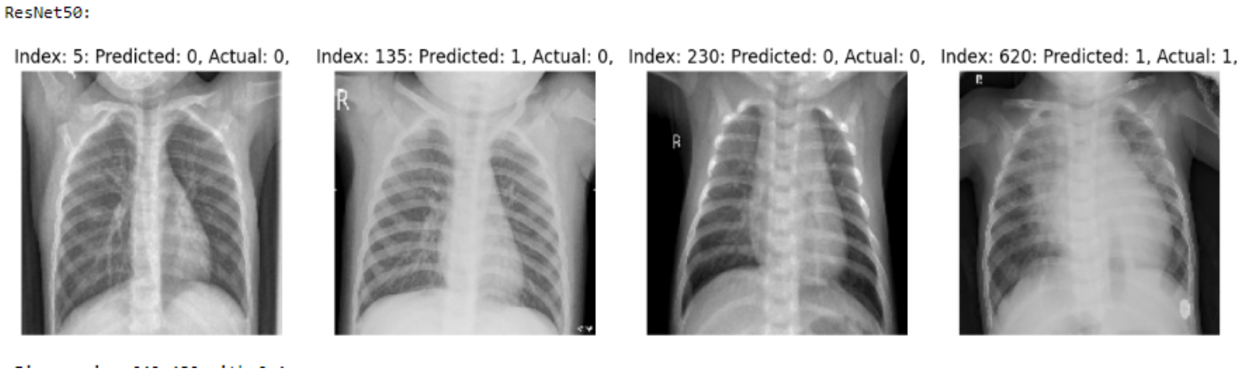
a. In the above ROC curve for ResNet50, we see that the ROC curve area is 0.92 ; which is a relatively high score and indicates that the ResNet50 model can effectively differentiate between the two classes; “Normal” and “Pneumonia”.



b. In the above ROC curve for VGG, we see that the ROC curve area is 0.89 ; which indicates that the VGG model can also effectively differentiate between the two classes; “Normal” and “Pneumonia”.

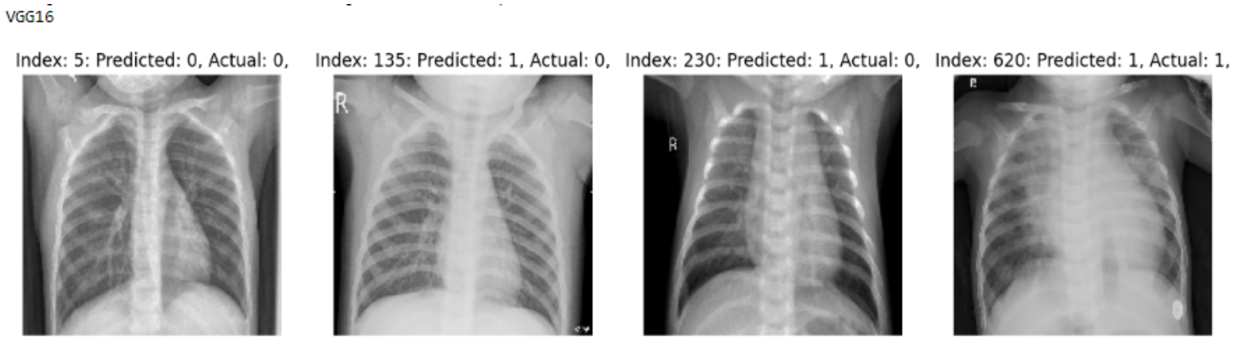
Examples

1. Example of Prediction by ResNet50:



```
ResNet50:

Index: 5: Predicted: 0, Actual: 0,   Index: 135: Predicted: 1, Actual: 0,   Index: 230: Predicted: 0, Actual: 0,   Index: 620: Predicted: 1, Actual: 1,
```

2. Example of Prediction by VGG16:



```
VGG16

Index: 5: Predicted: 0, Actual: 0,   Index: 135: Predicted: 1, Actual: 0,   Index: 230: Predicted: 1, Actual: 0,   Index: 620: Predicted: 1, Actual: 1,
```

Four Images were selected randomly, same set of images were used for prediction by both ResNet and VGG16 model,From the above images, we can construe that :

- Image with index 5 was correctly classified as class 0 by both ResNet and VGG16, so it is a true positive for both models
- Image with index 135 incorrectly classified as class 1 by both models, but actually belonged to class 0. Thus it is a false positive for both models
- Image with index 230, was correctly classified as class 0 by the ResNet model whereas it was incorrectly classified as class 1 by VGG16 model. Thus, it is true positive for ResNet whereas it is False positive for the VGG16 model.

- Image with index 620 was correctly classified as class 1 by both ResNet and VGG16, so it is a true negative for both models.

Based on the analysis, we observed that both ResNet and VGG16 models were able to correctly classify two of the images, resulting in true positives and true negatives for both models. However, both models misclassified one image, leading to false positives for both. Additionally, one image was correctly classified by ResNet but misclassified by VGG16, resulting in a true positive for ResNet but a false positive for VGG16.

Comparison of Models

1. **Accuracy :**
   After being trained for 10 epochs with a batch size of 20, ResNet50 achieved a higher training accuracy of 98.77% and validation accuracy of 89.5% compared to VGG16 which had a training accuracy of 97.71% and validation accuracy of 92%. The ResNet model has achieved a higher training accuracy than VGG, which means that the model is able to capture the training data well. On the other hand, VGG16 has achieved a higher validation accuracy, which means that the model is able to generalize well to unseen data.

2. **Performance:**
   According to the result of the evaluation, ResNet50 had an overall F1 score of 91% and VGG16 had an overall F1 score of 90.5%. The F1-score for both models is relatively high. The difference in F1 scores between the two models is relatively small.
   Comparing the ROC curves of ResNet50 and VGG16, we see that ResNet50 has a ROC curve area of 0.92 and VGG16 has a ROC curve area of 0.89, indicating that both ResNet50 and VGG models are good for classifying "Normal" cases and "Pneumonia" cases.
   The F1-Score and ROC-Curve of the ResNet50 model is slightly higher than the VGG model indicating that the ResNet50 model outperforms the VGG model in overall performance.

3. **Training Speed**
   Both models were trained for 10 epochs, with a a batch size of 20. VGG16 took more time to train than ResNet50, this is because VGG models typically have more parameters than ResNet50 models, which means that they require more computations and memory during training.Also, ResNet50 models have a unique architecture that allows for faster training by using

skip connections.

4. **Generalization**
   ResNet50 had a lower training loss of 0.0325 than VGG16 with a training loss of 0.0711. ResNet50 had a higher validation loss of 0.4102 than VGG16 which had a validation loss of 0.2841.ResNet50 had a higher validation loss than VGG16, which means that it did not generalize as well to new data as VGG16. However, ResNet50 achieved a higher validation accuracy than VGG16, which indicates that it was able to correctly classify more samples in the validation set despite having a higher validation loss.

5. **Complexity:**
   In terms of architecture and complexity, ResNet50 is more complex than VGG16 as ResNet50 has 50 layers; with a series of convolutional layers with residual blocks, shortcut connections, and global average pooling. The residual blocks allow for the flow of information directly from one layer to another, which makes the architecture deeper and more complex.VGG16 simply has 16 layers; a series of 13 convolutional layers and 3 fully connected layers.

Discovery

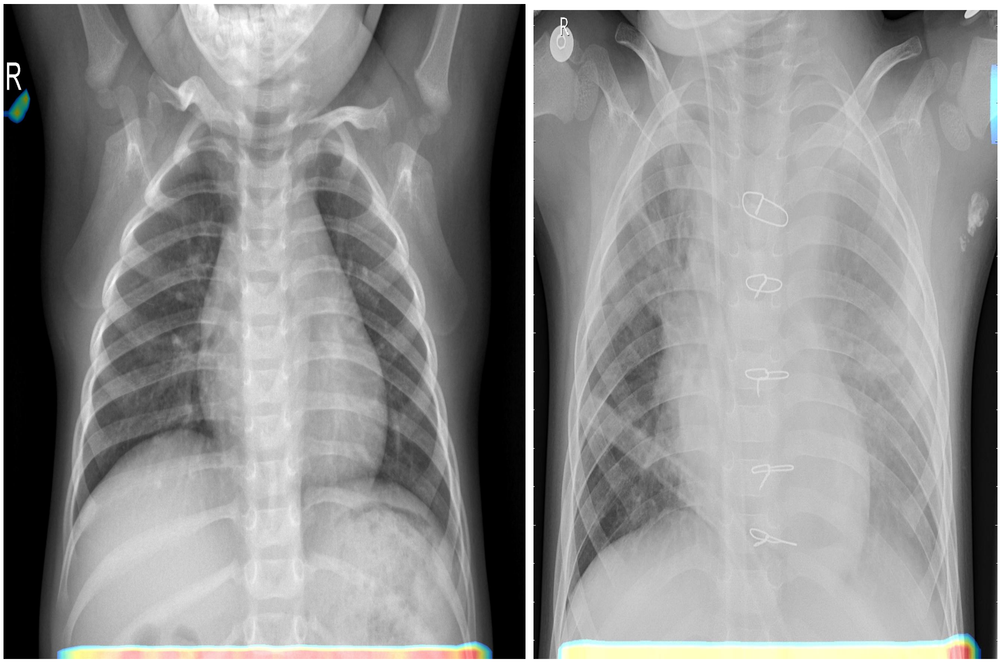
- **Model Performance**:
  The ResNet model achieved an overall accuracy of 91%.F1 score of 93% for the "Pneumonia" class indicates that the model was able to correctly identify most of the pneumonia cases, while the F1 score of 89% for "Normal" class suggests that the model was slightly less accurate in identifying normal cases.
  Similarly, the VGG model achieved an overall accuracy of 91%. The F1 score of 94% for the "Pneumonia" class indicates that the model was able to correctly identify most of the pneumonia cases, while the F1 score of 87% for the "Normal" class suggests that the model was slightly less accurate in identifying normal cases.
  In both models, the confusion matrix showed that the model had a higher number of false positives than false negatives, which means that the models were more likely to classify normal cases as pneumonia cases than vice versa.

- **Model Architecture and training speed:**
  ResNet50 has a deeper and more complex architecture compared to VGG16. ResNet50 consists of 50 layers, which include a series of convolutional layers with residual blocks, shortcut connections, and global average pooling. On the other hand, VGG16 has 16 layers, which consist of 13 convolutional layers and 3 fully connected layers.
  Based on the experiment, despite its complex architecture, ResNet50 has been found to have a faster training speed compared to VGG16. This is due to the use of shortcut connections that help alleviate the vanishing gradient problem, as well as its lower memory and computation requirements during training.

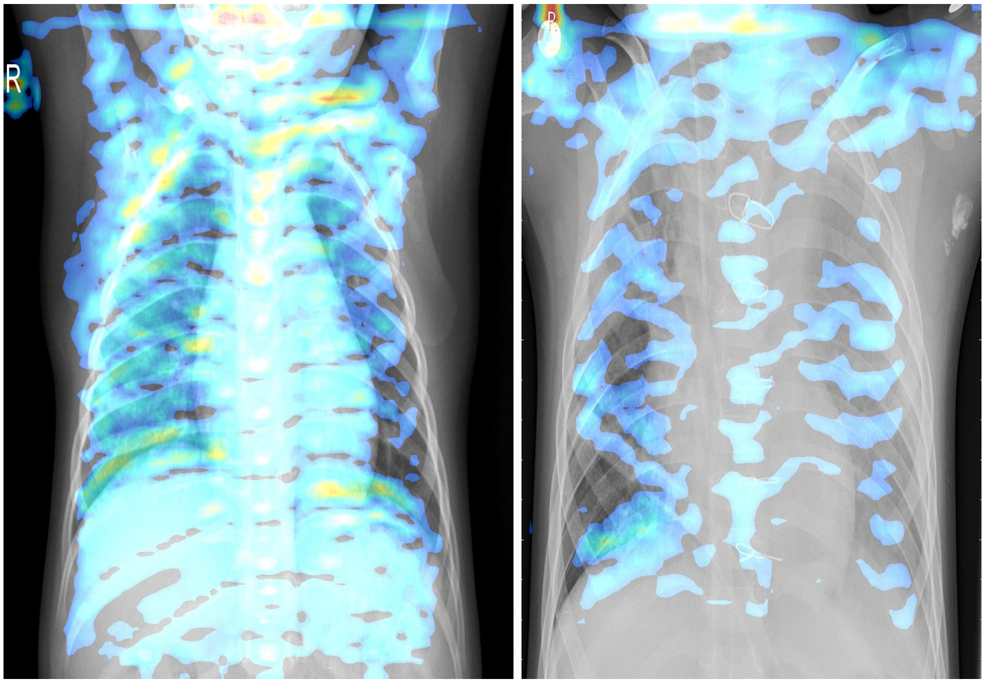
- **Working of Models:**
  In terms of identifying images, ResNet50 is found to be better than VGG16. This is because ResNet50 uses a special type of layer that allows information to flow more easily between different layers in the network. This means that ResNet50 can better recognize small and subtle details in the images. On the other hand, VGG16 uses many convolutional layers, which can make the network too specialized for the training data. This can cause the network to perform poorly on new data that it hasn't seen before.

- **Class Activation Maps:** Despite a better ability to recognize subtle details of images, ResNet fails to create better and more precise class activation images like VGG16.

  A class activation Map is an image generated by a neural network model that highlights the regions of an input image that the model used to make a particular classification decision. It helps in understanding which parts of an image were important for a given classification decision. The generated Class Activation Map images using the trained ResNet50 model and VGG16 model are shown below:



i. Class Activation Map using ResNet



ii. Class Activation Maps using VGG16

The reason ResNet50 does not produce clear and precise class activation maps like of VGG16 could be due to the complicated architecture of ResNet50. ResNet50 uses skip connections, which can help improve the accuracy of the model, but it can also make it harder to visualize the class activation maps. Also, ResNet50 has a larger number of parameters, which can make it more challenging to interpret the class activation maps as the model may be picking up on subtle features that are difficult for humans to identify.

On the other hand VGG16 has a simpler architecture, with only convolutional and pooling layers, which can make it easier to interpret the class activation maps.