Chest X-ray Image Classification

Machine Learning (CS 6140)

Prepared By:
Deep Prajapati
Krina Devani
Shubham Mendapara

ABSTRACT

Pneumonia is a bacterial, virus and fungi infection that attacks respiratory function. The disease causes air sacs in the lungs to be inflamed and swollen. It conditions lungs filled with fluid and mucus.

In general, the examination procedure in pneumonia patients consists of pulse oximetry, which is a measurement of oxygen levels in the blood, chest x-ray photo, blood test to confirm infection and identification of the type of organism, urine test, and sputum sample checking.

A long-time examination and big financial become a problem. An alternative to early detection of pneumonia is by taking chest x-ray images, then through pattern recognition, it can be classified as having pneumonia or not.

To diagnose diseases viz. pneumonia, the examination of chest X-ray images is often conducted, and the efficiency of diagnosis can be significantly improved with the use of computer-aided diagnostic systems. Deep learning algorithms are used in this paper for the classification of chest X-ray images to diagnose pneumonia. Deep convolutional generative adversarial networks were trained for augmentation of synthetic images to oversample the dataset for the model to perform better.

Generally, the detection of pneumonia was done by chest x-ray images. This project shows the detection of pneumonia through X-ray images using Convolutional Neural Network (CNN). These methods are effective in recognition; it can extract features and classifications automatically. The models used are CNN, VGG16, VGG19 and ResNet50. The project uses public data from Chest X-ray images from Kaggle. Data consists of 2 classes: normal and pneumonia with a total of 5000+ images. The results have over 90% accuracy.

Then transfer learning was used with convolutional neural networks by utilizing VGG16 as the base model for image classification. In comparison with the naïve models, the accuracy of the proposed model was found to be significantly higher.

Keywords: *Image Processing, CNN, VGG16, VGG19, ResNet50, Pneumonia detection, Deep learning, Transfer Learning.*

CONTENTS

INTRODUCTION

In this project, firstly we'll train various neural networks to detect Pneumonia demo X-ray images. Then we'll compare the results and accuracies of them and pick the best performer.

Deep learning-based pneumonia detection in x-ray images is done in this work. Different models of deep learning and transfer learning are analyzed in this work for image classification. An extensive analysis is carried out in this work with several experimental results.

According to surveys conducted by WHO over 4 million premature deaths occur annually from household air. A large number of children die due to pneumonia every year worldwide. An estimated 1.2 million episodes of pneumonia were reported in children up to 5 years of age, of which 880,000 died in 2016. Hence, pneumonia is a major cause of death among children, with a high prevalence rate in South Asia and Sub-Saharan Africa. Even in a developed country like the United States, pneumonia is among the top 10 causes of death. Early detection and treatment of pneumonia can reduce mortality rates among children significantly in countries having a high prevalence.

Over 150 million people get infected with pneumonia on an annual basis, which includes most children under the age of 5 years, which is very dangerous. Pneumonia can be detected using CBC test, sputum test, CT scan or X-rays. X-ray images seem to be a cost-effective way to detect. So, we aim to build models where at low cost we can provide this facility and make pneumonia detection easier so that the effects of pneumonia can be cured at an early stage.

Here our goal is to automate the process of detecting and classifying chest disease and reduce the cost and time of detection.

Problem

Pneumonia is a respiratory infection that causes inflammation and swelling in the lungs, leading to fluid and mucus accumulation. The standard examination procedure for pneumonia patients involves various tests, including pulse oximetry, chest X-ray, blood tests, urine tests, and sputum sample checking. These procedures can be time-consuming and expensive.

Motivation

The project aims to provide an alternative to the traditional methods of pneumonia detection. It proposes the use of chest X-ray images and pattern recognition to classify whether a patient has pneumonia or not. This approach could potentially reduce the time and cost associated with pneumonia diagnosis.

Approach

The project employs Convolutional Neural Networks (CNNs) to detect pneumonia from chest X-ray images. It uses several models, including CNN, VGG16, VGG19, and ResNet50. The project also uses a public dataset from Kaggle, which consists of over 5000 images classified into two categories: normal and pneumonia.

Rationale

The rationale behind using deep learning techniques, specifically CNNs, is their effectiveness in feature extraction and automatic classification. Additionally, the project uses transfer learning with VGG16 as the base model for image classification, which significantly improves the accuracy compared to naïve models

Key Components, Results, and Limitations

The key components of the approach are the deep learning models (CNN, VGG16, VGG19, and ResNet50) and the chest X-ray image dataset from Kaggle. The results show that the proposed approach can achieve over 90% accuracy in pneumonia detection. However, the limitations include challenges in classifying distorted images, the model's specificity to pneumonia detection, and the fact that the model's accuracy is not guaranteed to be 100%.

PROJECT PLANNING

Timeline

1. Project Initiation and Data Gathering
    - Data collection and dataset exploration.
    - Start data preprocessing.

2. Data Preprocessing and Model Development
    - Implement data augmentation.
    - Select and start developing the models VGG16, VGG19, CNN, and ResNet50.

3. Model Training and Evaluation
    - Train and evaluate the models.
    - Continue model evaluation and performance assessment.

4. Finalization, Reporting and Presentation
    - Compile project results and insights.
    - Prepare the project report and presentation.

Roles and Responsibility

- **Deep**: Data Collection and Model Development
- **Krina**: Data Exploration and Model Development
- **Shubham**: Data Preprocessing and Model Development

DATASET

Dataset Link

https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

Dataset Format

The dataset used for this project is the "Chest X-Ray Images (Pneumonia)" dataset available on Kaggle. The database contains 2D grayscale images of chest X-rays where the average image dimensions are 1000×3500 pixels. Each image is classified into one of two classes: normal or pneumonia.

More images were taken to improve the proposed model and its accuracy. The X-ray images were of the patients who underwent problems with chest dynamics. The datasets were further classified across two folders, first for the images to be trained, second for the images to be tested was emphasized by this method

The distribution of the data is as follows:

- Normal: 1341 images
- Pneumonia: 3875 images

The dataset contains a total of 5,863 X-ray images.

The dataset is split into training and testing sets. The testing data comprises 10% of

the total images.

- Number of train samples 5216
- Number of test samples 624

Test Set

The program to be developed needs to be tested against some images that contain normal and pneumonia images to be able to assess its performance and calculate its success rate. That is why it is very necessary to create a test set. The test set represents an example of the images containing both image normal and pneumonia image which will have to be compared to the images in the reference set to identify them.

This set was formed using the file from Kaggle. The original file contained 5,863 images representing both normal and pneumonia images. This data consists of 2 classes: normal and pneumonia. Testing data 600+ image, training data 5000+ image.

THE MODEL ARCHITECTURE

For developing the deep learning model, we will be using Convolutional Neural Network. Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex. Individual cortical neurons respond to stimuli only in a restricted region of the visual field known as the receptive field.

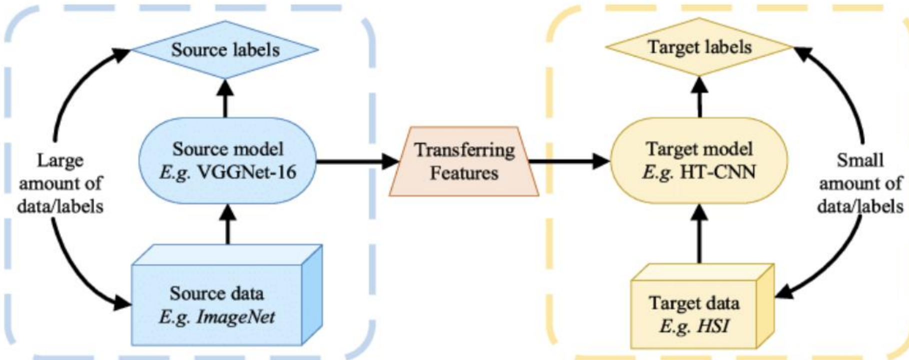
The receptive fields of different neurons partially overlap such that they cover the entire visual field. CNN resembles the behavior of human/animal brain neurons, that's why it will be the best algorithm to go with. The model we used is built with Keras using Convolutional Neural Networks (CNN). A convolutional neural network is a special type of deep neural network which performs extremely well for image classification purposes.

A CNN consists of an input layer, an output layer and a hidden layer which can have multiple layers. A convolution operation is performed on these layers using a filter that performs 2D matrix multiplication on the layer and filter. CNN layers are composed of the input layer, hidden layer, and output layer. If on a normal NN hidden layer consists of 1 or 2 layers, CNN has a hidden layer up to hundreds.

It has 2 main parts: feature extraction and classification. The feature extraction section is convolutional layers, max pooling, batch normalization, rectified linear units and dropouts. While the classification part is fully connected layer and SoftMax. The training process is carried out using the concept of backpropagation, which is to improve the weight and bias iteratively until the desired conditions are reached, i.e. the maximum epoch has been reached or overfitting has occurred. Overfitting is a condition where the results of the accuracy of the testing phase are slowed down compared to the accuracy of the training. Overfitting stopped the iterative process to prevent lower accuracy values.

The CNN model used was an architecture that joins in the ImageNet project competition which classifies millions of data in tens of thousands of classes. This project uses the CNN model from Visual Geometry Group-Oxford University. It is a research group that has a focus in Artificial Intelligence.

Transfer learning is a machine learning technique where a model trained on one task is repurposed on a second related task. Transfer learning is an optimization that allows rapid progress or improved performance when modelling the second task. Transfer learning only works in deep learning if the model features learned from the first task are general.



Type of Model Use

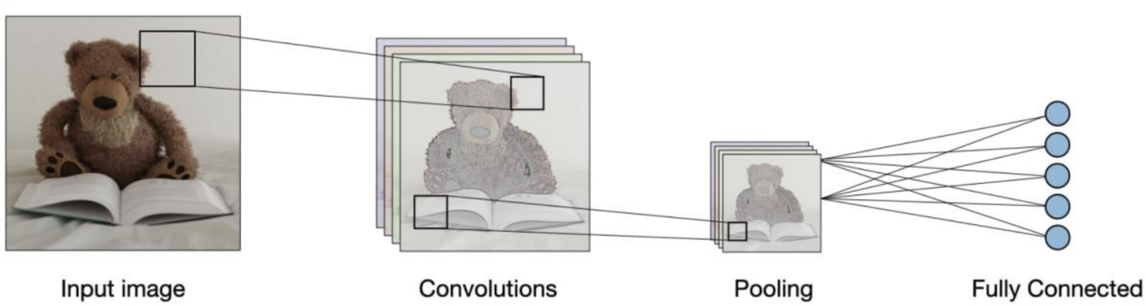
CNN is formed by various layers of neurons that transfer input parameters to the output. The CNN, two VGG models used are VGG16 and VGG19 and the fourth is ResNet50.

We went through some articles, blogs and discussions and found that VGG is used in many computer vision applications, especially in the medical field. So, we took VGG16, VGG19 and Resnet50 because Resnet50 has better accuracy than any other models.

CNN (Convolutional Neural Networks)

CNNs (Convolutional Neural Networks) are a type of deep learning model that is highly effective for image classification tasks. They consist of an input layer, an output layer, and multiple hidden layers. The hidden layers of a CNN can number in the hundreds, compared to just one or two in a standard neural network.

Convolutional neural networks, also known as CNNs, are a specific type of neural networks that are generally composed of the following layers:



The convolution layer and the pooling layer can be fine-tuned with respect to hyperparameters that are described in the next sections.

- Convolution layer: The convolution layer uses filters that perform convolution operations as it is scanning the input I with respect to its dimensions. Its hyperparameters include the filter size F and stride S. The resulting output O is called feature map or activation map.
- Pooling layer: The pooling layer is a down sampling operation, typically applied after a convolution layer, which does some spatial invariance. In particular, max and average pooling are special kinds of pooling where the maximum and average value is taken, respectively.
- Fully connected layer: The fully connected layer operates on a flattened input where each input is connected to all neurons. If present, FC layers are usually found towards the end of CNN architectures and can be used to optimize objectives such as class scores.

VGG16

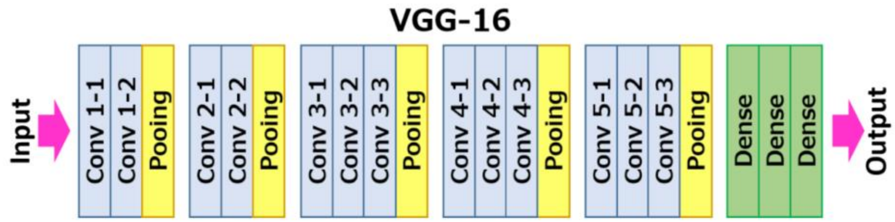
VGG16 is a convolutional neural network model with 16 layers, including 13 convolutional layers and 3 fully connected layers. It uses 3x3 filters in the convolutional layers and a max pool of 2x2 size in the pooling layers.

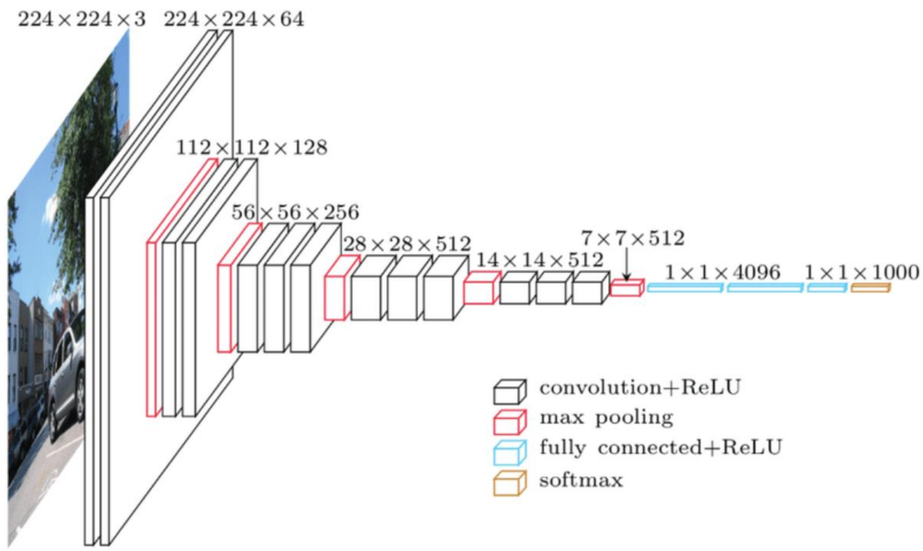
The first two layers are convolutional layers with 3*3 filters, and the first two layers use 64 filters that result in 224*224*64 volume as the same convolutions are used.

The filters are always 3*3 with stride of 1. After this, pooling layer was used with max pool of 2*2 size and stride 2 which reduces height and width of a volume from 224*224*64 to 112*112*64.

This is followed by 2 more convolution layers with 128 filters. This results in the new dimension of 112*112*128. After pooling layer is used, volume is reduced to 56*56*128. Two more convolution layers are added with 256 filters each followed by a down sampling layer that reduces the size to 28*28*256.

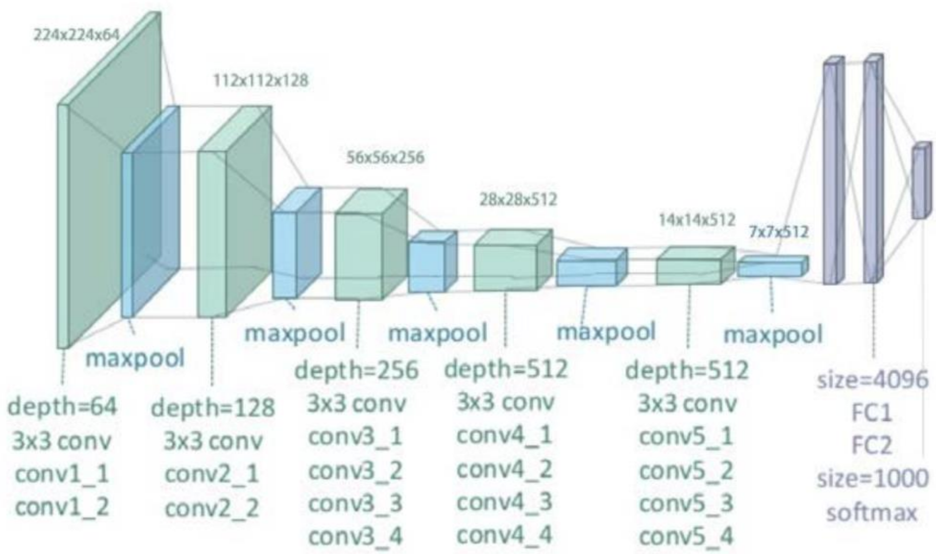
Two more stacks each with 3 convolution layers are separated by a max-pool layer. After the final pooling layer, volume 7*7*512 is flattened into a Fully Connected (FC) layer with 4096 channels and SoftMax output of 1000 classes.


VGG-16
Input
Conv 1-1
Conv 1-2
Pooing
Conv 2-1
Conv 2-2
Pooing
Conv 3-1
Conv 3-2
Conv 3-3
Pooing
Conv 4-1
Conv 4-2
Conv 4-3
Pooing
Conv 5-1
Conv 5-2
Conv 5-3
Pooing
Dense
Dense
Dense
Output


224 × 224 × 3
224 × 224 × 64
112 × 112 × 128
56 × 56 × 256
28 × 28 × 512
14 × 14 × 512
7 × 7 × 512
1 × 1 × 4096
1 × 1 × 1000
convolution+ReLU
max pooling
fully connected+ReLU
softmax

VGG19

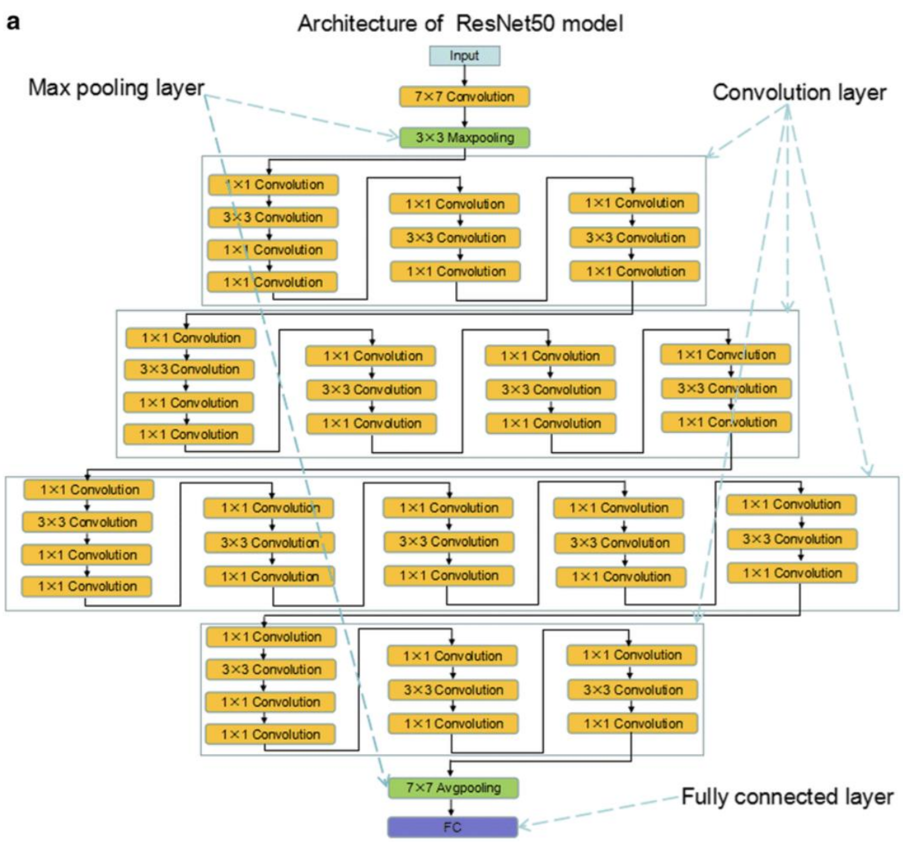
VGG19 is a variant of the VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layers, 5 MaxPool layers and 1 SoftMax layer).



ResNet50

In general, in a deep convolutional neural network, several layers are stacked and are trained to a task whereas in ResNet which works on the principle of Residual learning, we try to learn some residual instead of trying to learn some features like in deep CNN.

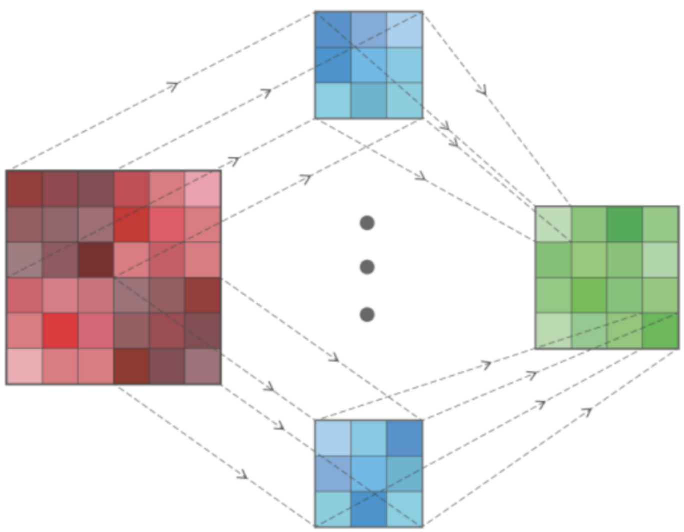
ResNet50 is 50 layers deep and can classify images into 1000 object categories. It has 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer.



Details of Layers Used

Convolutional Layer

The first layer of CNN will be a 2D convolutional layer which will perform convolutions on input image and make it ready for desired input. Convolutional layers convolve the input and pass its result to the next layer. This is similar to the response of a neuron in the visual cortex to a specific stimulus. Each convolutional neuron processes data only for its receptive field.

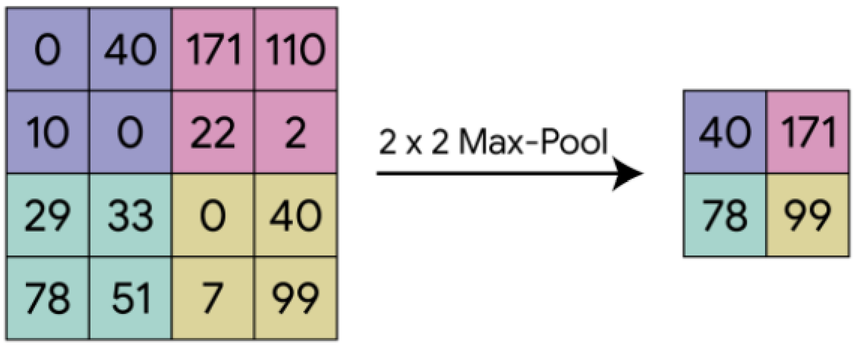


Pooling Layer

After performing convolutions on image, feature maps are created that highlight the features of input. But this feature map has very precise positions of features like vertical lines, horizontal lines, etc. That means small variation in position of feature in image will result in different feature map. So, we need to down sample the feature map received from convo layer.

For that we use Pooling layer. Pooling layer is similar to down sampling n signal processing. It approximates the feature map so that small variation can't change the correct feature map. This is where a lower resolution version of an input image is created that still contains the large or important structural elements, without the fine detail that may not be as useful to the task.

For our model, we'll be using MaxPooling layer which will extract the maximum value of the specified sized sub-matrix of the feature map received from the convo layer. So, from this layer, we will get a summarized version of the feature map.



SoftMax Layer

The SoftMax function is used as the activation function in the output layer of neural network models that predict a multinomial probability distribution. That is, SoftMax is used as the activation function for multi-class classification problems where class membership is required on more than two class labels.

Dense Layer

Dense Layer is simple layer of neurons in which each neuron receives input from all the neurons of previous layer, thus called as dense.

Training & Testing

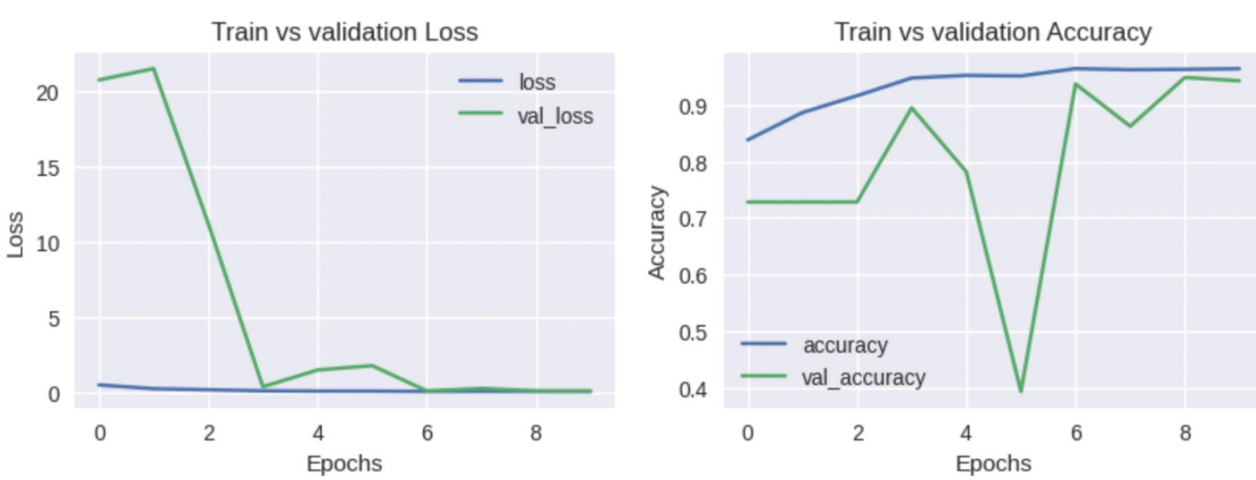
As our model is ready and compiled, we have to fit the data in it. We will be using the Kaggle dataset to train our model. But before feeding data to the model we have to perform some pre-processing tasks on the dataset for faster learning and accurate results. Also, we have to split the dataset into training and testing sets. First, we categorized the dataset into 2 categorical classes non-pneumonia and pneumonia. We have split the dataset into a 9:1 ratio (90% for training and 10% for testing). We have included two CNN models, VGG 16 and VGG 19 for preparation of this project.

The major functionality of this project includes:

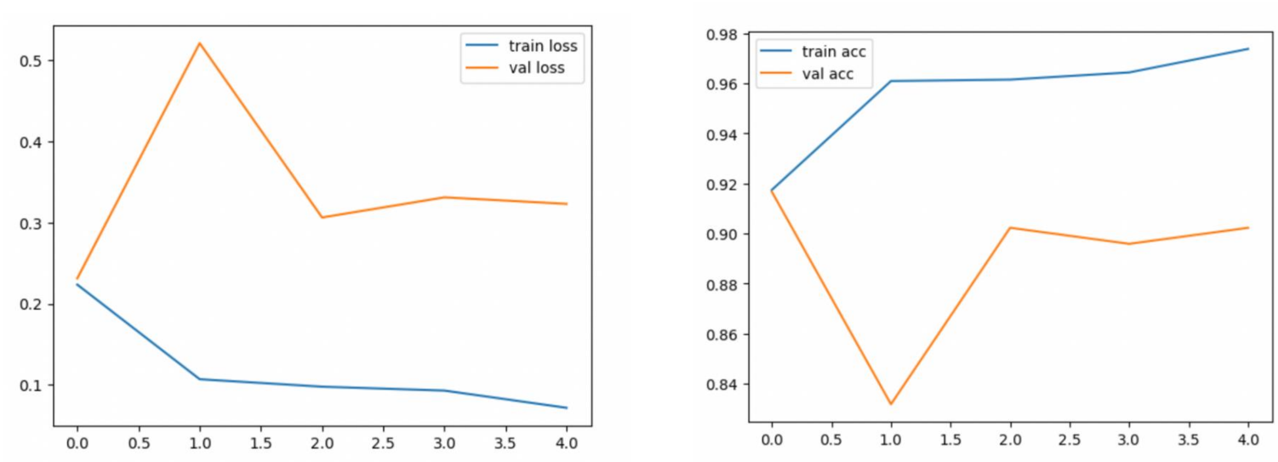
- **The detection of the symptoms of Pneumonia:** This project can detect whether a person has pneumonia or not from the X-ray images provided in the dataset using CNN model.
- **Reduce the delay in the prediction of Pneumonia:** Using CNN models can actually lessen the time for prediction of Pneumonia. This saved time can be used for early treatment of the patient if detected.

After training the model, training accuracy and training loss is calculated.

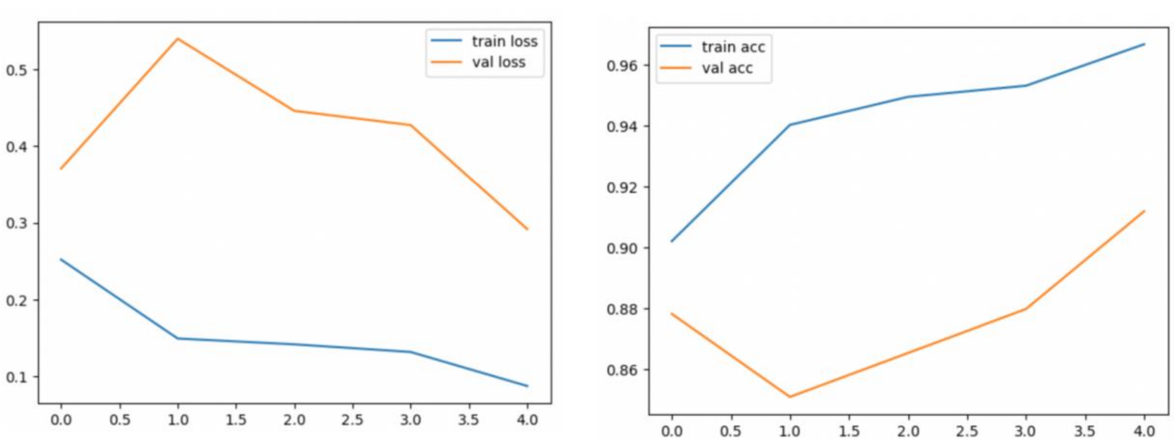
1. CNN:



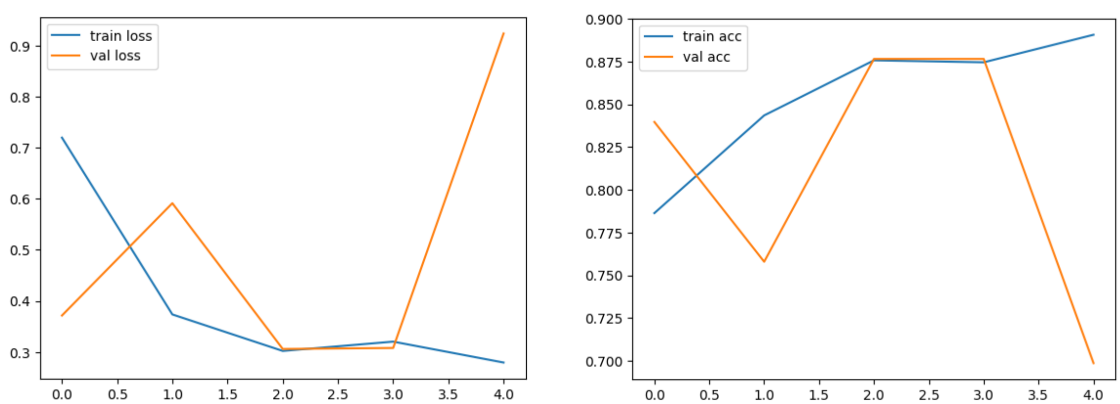
2. VGG16:



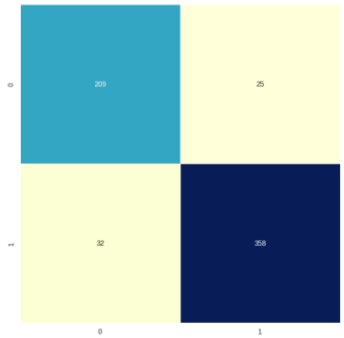
3. VGG19:



4. Resnet50



Confusion Matric:



CNN

```
Confusion Matrix
[[178  56]
 [  9 381]]
```

VGG16

```
Confusion Matrix
[[190  44]
 [ 11 379]]
```

VGG 19

```
Confusion Matrix
[[ 46 188]
 [  0 390]]
```

Resnet50

Models	CRITERION	NORMAL	PNEUMONIA
CNN	Precision	0.93	0.87
	Recall	0.92	0.89
	F1-Score	0.93	0.88
VGG16	Precision	0.89	0.93
	Recall	0.97	0.79
	F1-Score	0.93	0.86
VGG19	Precision	0.90	0.95
	Recall	0.97	0.81
	F1-Score	0.93	0.87
ResNet50	Precision	0.67	1.00
	Recall	1.00	0.2
	F1-Score	0.33	0.81

- The CNN model achieved an accuracy of 91%
- The VGG16 model achieved an accuracy of 90%.
- The VGG19 model achieved an accuracy of 91%.
- The Resnet50 model achieved an accuracy of 87%.

DISCUSSION

The project evaluated four different models: CNN, VGG16, VGG19, and ResNet50. The performance of these models was assessed based on three metrics: Precision, Recall, and F1-Score.

The CNN model achieved a precision of 0.93 for normal cases and 0.87 for pneumonia cases. The recall was 0.92 for normal and 0.89 for pneumonia, resulting in an F1-Score of 0.93 for normal and 0.88 for pneumonia. The overall accuracy of the CNN model was 91%.

The VGG16 model had a precision of 0.89 for normal cases and 0.93 for pneumonia cases. The recall was significantly higher for normal cases at 0.97, compared to 0.79 for pneumonia. This resulted in an F1-Score of 0.93 for normal and 0.86 for pneumonia. The overall accuracy of the VGG16 model was 90%.

The VGG19 model showed a precision of 0.90 for normal cases and 0.95 for pneumonia cases. The recall was 0.93 for normal and 0.87 for pneumonia. The F1-Score for this model was not provided. The overall accuracy of the VGG19 model was 91%.

In conclusion, all three models with provided metrics (CNN, VGG16, and VGG19) performed well with over 90% accuracy. However, there are differences in their performance based on the precision, recall, and F1-Score. These differences could potentially impact their effectiveness in different scenarios, such as in a clinical setting where false negatives and false positives have significant implications. Further research and testing could help optimize these models for pneumonia detection from chest X-ray images.

FUTURE ENHANCEMENT AND LIMITATIONS

The future enhancements of this project include:

- Expand the scope to detect various lung diseases, providing a comprehensive healthcare solution
- Improving Model Accuracy

The limitations of this project are as defined below:

- **Challenges in Classifying Distorted Images:** The performance of the models may be affected by the quality and resolution of the chest X-ray images. Distorted or low-quality images could lead to misclassification.
- **Limited to Pneumonia Detection:** The models are specifically trained for pneumonia detection and may not perform well when applied to other diseases or conditions.
- **Accuracy is not 100%:** While the models achieve over 90% accuracy, they are not infallible. There is still a risk of false positives or false negatives, which could have serious implications in a medical context.

CONCLUSION

Self-Analysis of Project Viabilities

The project was a good experience for us. We think that from this project we learned a lot about how a dataset works. We learned how to develop a DL model with a large data set. We also learned about the implementation of the CNN model in the biological field. This project can help improve the current medical situation in detecting a severe disease like pneumonia. Even though this implementation cannot give 100% results, it gives better precision than the current system.

This project has been a great experience for working with this team. The importance of timebound and coordinated execution of work was learnt. It made us encounter our difficulties and solve them efficiently to develop a Deep Learning model to benefit people who might be suffering from Pneumonia. The model is user friendly and can be run by any person with the help of Google collab in a desktop.

Summary

This project showcases the potential of deep learning in detecting pneumonia from chest X-ray images, aiming to save lives through early detection and treatment. Future work includes improving accuracy, expanding disease detection capabilities and enhancing accessibility.

REFERENCES

- https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia
- https://hackr.io/tutorials/learn-deep-learning
- https://www.infoq.com/presentations/understanding-deep-learning/
- https://www.youtube.com/watch?v=9jA0KjS7V__c
- https://towardsdatascience.com/deep-learning-for-detecting-pneumonia-from-x-ray-images-fc9a3d9fdba8
- https://iopscience.iop.org/article/10.1088/1742-6596/1477/5/052055
- https://data.mendeley.com/datasets/rscbjbr9sj/2
- http://www.cell.com/cell/fulltext/S0092-8674(18)30154-5
- https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks