**LUNG DISEASE PREDICTION USING DEEP LEARNING**

## Project Report

submitted in partial fulfillment for the award of the degree of

#### BACHELOR OF TECHNOLOGY

**in**

#### COMPUTER SCIENCE AND ENGINEERING

By

G. Revanth Kumar (181FA04473)

**Under the Supervision of**

#### Mr.Saiyed Faiyaz Waris,

Assistant Professor, CSE



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING VIGNAN’S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH (Deemed to be UNIVERSITY)**

-Est u/s 3 of UGC act 1956 VADLAMUDI :: GUNTUR – 522213 ANDHRA PRADESH :: INDIA

#### JUNE 2022

**VIGNAN’S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH (Deemed to be UNIVERSITY)**

#### Department of Computer Science and Engineering



**CERTIFICATE**

This is to certify that the project report entitled “**LUNG DISEASE PREDICTION USING DEEP LEARNING**” is the bona fide record of project work carried out under my supervision by **G. Revanth Kumar (181FA04473),** during the academic year 2021-2022, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering of Vignan’s Foundation for Science, Technology and Research (Deemed to be University), Guntur. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

**Head of the Department Signature of Project Guide**

**(Dr. VENKATESULU DONDETI) (Mr.Saiyed Faiyaz Waris)**

Professor Assistant professor

**DECLARATION**

I hereby declare that the project report entitled “**LUNG DISEASE PREDICTION USING DEEP LEARNING**” has been written by us and has not been submitted either in part or whole for the award of any degree, diploma or any other similar title to this or any other university .

1. G. Revanth Kumar (181FA04473)

Date: 22/06/2022

Place: Guntur

**ACKNOWLEDGEMENT**

It gives me a great sense of pleasure to acknowledge the assistance and cooperation I have received from several persons while undertaking this B. Tech. Final Year Project. I owe special debt of gratitude to **Mr. Saiyed Faiyaz Waris** Department of Computer Science & Engineering, for his constant support and guidance throughout the course of my work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for me.

I also take the opportunity to acknowledge the contribution of **Prof. Dr. VENKATESULU DONDETI,** Head, Department of Computer Science & Engineering, for his full support and assistance during the development of the project.

I also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

G. Revanth Kumar (181FA04473)

.

**ABSTRACT**

#### Nowadays for identifying or predicting any diseases on human beings, we should have a proper diagnosis for predicting the disease which is present in that human body. In general for prediction of diseases we try to use X-Ray, CT or MRI scan techniques for taking decisions on that appropriate disease. In general medical people need complete knowledge on that appropriate domain to find out the abnormality which is present in human beings. As we all know that India tops the world for having more deaths due to lung diseases. After the second highest cause of deaths in India due to heart disease, this long disease is one which is increasing its rank more and more. In order to reduce that problem early diagnosis and treatment of lung diseases is critical to prevent complications including death. Normally for finding the abnormality present in lungs, chest X-ray is playing a very important role to detect the complete information about the lungs. In this current article we try to present an effective way for expert diagnosis of lung diseases using deep learning models. It focuses on creating a system for assistance of Radiologists in detection of lung diseases. This will especially benefit rural areas where radiologists aren’t easily available. We use a CNN model called VGG 16 for predicting the lung disease from chest X ray images and then tell accuracy and performance of the proposed model. We conclude by discussing research obstacles, emerging trends, and possible future directions for improving some more advancement.

**TABLE OF CONTENTS**

**DECLARATION…………………………………………………………………...i**

**ACKNOWLEDGMENT…………………………………………………………..ii**

**ABSTRACT………………………………………………………………………..iii**

**LIST OF FIGURES……………………………………………………………….vi**

**ABBREIVATIONS……………………………………………………………….vii**

**INTERNSHIP SUMMARY……………………………………………………...viii**

**PROFILE OF THE COMPANY………………………………………………....ix**

**Chapter-1:INTRODUCTION**

1.1 INTRODUCTION……………………………………………………...1

1.2 IMAGE CLASSIFICATION…………………………………………...2

1.3 PROCESS TO APPLY DEEP LEARNING …………………………...3

1.3.1 [IMAGE ACQUISITION](#_heading=h.3rdcrjn)……………………………………………3

1.3.2 PREPOCESSING PHASE………………………………………….4

1.3.3 TRAINING PHASE………………………………………………...4

1.3.4 CLASSIFICATION PHASE………………………………………..4

1.4 IMAGE CLASSIFIACTION…………………………………………...5

**Chapter-2:LITERATURE SURVEY**

2.1 [PROBLEM STATEMENT](#_heading=h.26in1rg)…………………………………………….8

2.2 EXISITING SYSTEM AND ITS LIMITATIONS…………………….8

2.3 PROPOSED SYSTEM…………………………………………………8

2.4 SYSTEM REQUIREMENTS…………………………………………..9

**Chapter-3:**[**DESIGN**](#_heading=h.lnxbz9)

3.1 [MODEL ARCHITECTURE…………………………………………..11](#_heading=h.26in1rg)

3.2 COMPONENTS IN ARCHITECTURE…………………………….12-13

**Chapter-4:**[**IMPLEMENTATION**](#_heading=h.lnxbz9)

4.1 [METHODOLOGY DESCRIPTION](#_heading=h.26in1rg)…………………………………..15

4.1.1 [CONOVOLUTIONAL LAYER](#_heading=h.3rdcrjn)…………………………………..15

4.1.2 RECTIFIED LINEAR UNIT LAYER…………………………….15

4.1.3 POOLING LAYER………………………………………………..15

4.1.4 FULLY CONNECTED LAYER………………………………….16

**Chapter-5:**[**RESULTS**](#_heading=h.lnxbz9)

5.1 STEP WISE CODE PROCESS………………………………………18

5.2 SOURCE CODE……………………………………………………..19

5.3 RESULTING IMAGES…………………………………………..22-24

**Chapter-6:CONCLUSION AND FUTURE WORK**

6.1 CONCLUSION………………………………………………………26

6.2 ISSUES AND FUTURE DIRECTION OF PROJECT………………26

6.2.1 ISSUES……………………………………………………….26-27

6.2.2 POTENTIAL FUTURE WORK……………………………...27-28

**Chapter-7:REFERENCES**

**LIST OF FIGURES**

* **FIGURE 1: TYPE OF EXAMINATIONS.**
* **FIGURE 2: VGG 16 MODEL ARCHITEATURE.**
* **FIGURE 3: VGG 16 LAYERS.**
* **FIGURE 4: VGG16 LAYERS THROUGH TRANSFER LEARNING.**
* **FIGURE 5: TRAINING LOSS DATA GRAPH.**
* **FIGURE 6: TRAINING VALIDATION ACCURACY GRAPH.**
* **FIGURE 7:PREDICTION OF DISEASE.**

**LIST OF ABBREVIATIONS**

* CNN : Convolutional Neural Networks
* VGG 16 : Visual Geometry Group
* RELU : Rectified Linear Unit
* Conv : Convolutional
* RGB : Red Green Blue
* CLAHE : Contrast Limited Adaptive Histogram Equalization
* ROI : Region of Interest
* DBN : Deep Belief Network
* MPNN : Multilayer perception neural network
* RNN : Recurrent neural network
* ILSVRC : ImageNet Large-Scale Visual Recognisation Challenge

**INTERNSHIP SUMMARY**

**Location:** Hyderabad

**Center:** TCS Ninja

**Duration:** 3 Months

**Date of Start:** 7 February, 2022

**Date of Submission:** 30 April, 2022

**Title of project:** Lung Disease prediction using Deep Learning

**Team Members:** G. Revanth Kumar-(181FA04473)

**Name of the Guide:** G. Surendhra Reddy

**Name of Faculty Guide:**Mr.Saiyed Faiyaz Waris, Associate Professor, CSE,VFSTR University

**Project Area:** Data Science

**Abstract:** The project predicting disease using lung scan is based on deep learning technology. The recent developments of deep learning support the identification and classification of lung diseases in medical images. Here we are creating a deep learning model based on the X-Ray scanning pictures by using CNN based VGG-16 model to predict the lung diseases.

**PROFILE OF THE COMPANY**

**About TCS**

**Tata Consultancy Services** (**TCS**) is an Indian multinational information technology (IT) services and consulting company with its headquarters in Mumbai. It is a part of the Tata Group and operates in 149 locations across 46 countries.

TCS is the second largest Indian company by market capitalisation and is among the most valuable IT service brands worldwide. In 2015, TCS was ranked 64th overall in the *Forbes* "World's Most Innovative Companies" ranking, making it one of the highest-ranked IT services companies and a top Indian company. As of 2018, it is ranked eleventh on the Fortune India 500 list.In April 2018, TCS became the first Indian IT company to reach $100 billion in market capitalisationand the second Indian company ever (Reliance Industries achieved it in 2007) after its market capitalisation stood at [₹](https://en.wikipedia.org/wiki/Indian_rupee)6.793 trillion (equivalent to ₹7.7 trillion or US$100 billion in 2020) on the Bombay Stock Exchange.

In 2016–2017, parent company Tata Sons owned 72.05% of TCS and more than 70% of Tata Sons' dividends were generated by TCS. In March 2018, Tata Sons sold stocks of TCS worth $1.25 billion in a bulk deal. As of 15 September 2021, TCS has recorded a market capitalisation of US$200 billion, making it the first Indian IT firm to do so.

**Services**

TCS offers a consulting-led, cognitive-powered, integrated portfolio of business, technology, and engineering services and solutions. This is delivered through its unique Location Independent Agile delivery model, recognized as a benchmark of excellence in software development.

**Address**

TATA CONSULTANCY SERVICES, SYNERGY PARK – CMC NON SEZ CENTRE,  
Old Mumbai Highway, Gachibowli, Hyderabad, Telangana.  
Land mark: Opposite to DLF building  
Hyderabad 500019, Telangana

# *Chapter 1*

***INTRODUCTION***

## 1. INTRODUCTION

## 1.1 Introduction

## In recent days, the introduction of IT and e-health care systems in the medical field are trying to provide medical experts to give proper treatment for the patients who are in emergency. One of the most critical diseases which is ranked second in India after heart disease is lung diseases, also known as respiratory diseases. Some of the diseases which come under respiratory disease are pneumonia, tuberculosis and currently Coronavirus Disease 2019 (COVID-19). As per the IRS ( International Respiratory Societies)report more than three hundred million people are continuously suffering from asthma disease and more than 2 million people die due to this lung disease. From the recent analysis, we know the COVID-19 pandemic infected millions of people and healthcare systems and also there was great loss for the humans. In general these lung diseases are major cause of death and create disaster for the world. Normally early detection of lung disease plays a major role in the chance of disease recovery and there are very few recovery rates if they are early detected and treated. In the primitive days the lung diseases are detected via blood test, skin test and some X-ray and CT scan. The report need to be examined by the radiology department and the concern person who has enough knowledge will try to tell the report from the test sample, which is becoming a very complex task if the radiologist is not available all the time. Recently deep learning has gained a lot of user’s attention towards medical domain for disease prediction and finding abnormality. Hence we try to use this deep learning technique on lung disease prediction and try to classify the abnormality which is present in the lungs using chest X-ray examination.

### 

### Figure 1: Types of Examinations

### From the above figure 1, we can clearly identify several types of examinations are done for identifying the abnormality which is present in human lungs. In general we try to apply deep learning in the field of medical domain to identify the pattern which is present in the chest X-ray and then try to derive the possible learned features from that image. As we all know that deep learning is becoming state of the art by increasing its performance in huge number of medical applications which can assist the medical department persons or clinicians to detect and classify some minute medical abnormalities very effectively and efficiently. There was a lot of research work undergone for the lung diseases detection and to the best of our knowledge we can see one survey paper which is published based on some previous published papers references on this topic. If we look in this paper we can see all the history related to deep learning and how deep learning is integrated in the applications of pulmonary imaging. Major applications of deep learning techniques on several lung diseases, namely pulmonary nodule diseases, pulmonary embolism, pneumonia, and interstitial lung disease, are also described.

### 1.2 Image Classification

### Image classification is the process of segmenting images into different categories based on their features. A feature could be the edges in an image, the pixel intensity, the change in pixel values, and many more. The biggest challenge when working with images is the uncertainty

### of these features like shape, size, color etc. To the human eye, it looks all the same however, when converted to data you may not find a specific pattern across these images easily. An image consists of the smallest indivisible segments called pixels and every pixel has a strength often known as the pixel intensity. Whenever we study a digital image, it usually comes with three color channels, i.e. the Red-Green-Blue channels, popularly known as the “RGB” values. Because RGB has seen that a combination of these three can produce all possible color pallets. Whenever we work with a color image, the image is made up of multiple pixels with every pixel consisting of three different values for the RGB channels. Now if we take multiple such images and try to label them as different individuals we can do it by analyzing the pixel values and looking for patterns in them. However, the challenge here is that since the background, the color scale, the clothing, etc. vary from image to image, it is hard to find patterns by analyzing the pixel values alone. Hence we might require a more advanced technique that can detect these edges or find the underlying pattern of different features in the face using which these images can be labeled or classified. This is where more advanced techniques like CNN come into picture.

### 1.3 The basic process to apply deep learning in lung disease prediction

### The process of how deep learning is applied to identify lung diseases from medical images is described. There are mainly three steps: image preprocessing, training and classification. Lung disease detection generally deals with classifying an image into healthy lungs or disease-infected lungs. The lung disease classifier, sometimes known as a model, is obtained via training. Training is the process in which a neural network learns to recognize a class of images. Using deep learning, it is possible to train a model that can classify images into their respective class labels. Therefore, to apply deep learning for lung disease detection, the first step is to gather images of lungs with the disease to be classified. The second step is to train the neural network until it is able to recognize the diseases. The final step is to classify new images. Here, new images unseen by the model before are shown to the model, and the model predicts the class of those images.

#### 1.3.1 Image Acquisition Phase

The first step is to acquire images. To produce a classification model, the computer needs to learn by example. The computer needs to view many images to recognize an object. Other types of data, such as time series data and voice data, can also be used to train deep learning models. In the context of the work surveyed in this project, the relevant data required to detect lung disease will be images. Images that could be used include chest X-ray, CT scan, sputum smear microscopy and histopathology image. The output of this step is images that will later be used to train the model.

#### 1.3.2. Preprocessing Phase

### The second step is preprocessing. Here, the image could be enhanced or modified to improve image quality. Contrast Limited Adaptive Histogram Equalisation (CLAHE) could be performed to increase the contrast of the images. Image modification such as lung segmentation and bone elimination could be used to identify the region of interest (ROI), whereby the detection of the lung disease can then be performed on the ROI. Edge detection could also be used to provide an alternate data representation. Data augmentation could be applied to the images to increase the amount of available data. Feature extraction could also be conducted so that the deep learning model could identify important features to identify a certain object or class. The output of this step is a set of images whereby the quality of the images is enhanced, or unwanted objects have been removed. The output of this step is images that were enhanced or modified that will later be used in training.

### 1.3.3. Training Phase

### In the third step, namely training, three aspects could be considered. These aspects are the selection of deep learning algorithm, usage of transfer learning and usage of an ensemble. There are numerous deep learning algorithms, for example deep belief network (DBN), multilayer perceptron neural network (MPNN), recurrent neural network (RNN) and the aforementioned CNN. Different algorithms have different learning styles. Different types of data work better with certain algorithms. CNN works particularly well with images. Deep learning algorithm should be chosen based on the nature of the data at hand. Transfer learning refers to the transfer of knowledge from one model to another. Ensemble refers to the usage of more than one model during classification. Transfer learning and ensemble are techniques used to reduce training time, improve classification accuracy and reduce over fitting. The output of this step is models generated from the data learned.

### 1.3.4. Classification Phase

### In the fourth and final step, which is classification, the trained model will predict which class an image belongs to. For example, if a model was trained to differentiate X-ray images of healthy lungs and tuberculosis-infected lungs, it should be able to correctly classify new images (images that are never seen by the model before) into healthy lungs or tuberculosis-infected lungs. The model will give a probability score for the image. The probability score represents how likely an image belongs to a certain class. At the end of this step, the image will be classified based on the probability score given to it by the model.

### 1.4 ImageNet Dataset

### ImageNet is a dataset of over 15 million labeled high-resolution images belonging to roughly 22,000 categories. The images were collected from the web and labeled by human labelers using Amazon’s Mechanical Turk crowd-sourcing tool. Starting in 2010, as part of the Pascal Visual Object Challenge, an annual competition called the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) has been held. ILSVRC uses a subset of ImageNet with roughly 1000 images in each of 1000 categories. At all, there are roughly 1.2 million training images, 50,000 validation images, and 150,000 testing images. ImageNet consists of variable-resolution images. Therefore, the images have been down-sampled to a fixed resolution of 256×256. Given a rectangular image, the image is rescaled and cropped out the central 256×256 patch from the resulting image.

# *Chapter 2*

***LITERATURE SURVEY***

**2. LITERATURE SURVEY**

Literature survey is that the most vital step in the software development process. Before developing the new application or model, it's necessary to work out the time factor, economy, and company strength. Once all these factors are confirmed and got approval then we can start building the application. The literature survey is one that mainly deals with all the previous work which is done by several users and what are the advantages and limitations of those previous models. This literature survey is mainly used for identifying the list of resources to construct this proposed application. 1) A Survey of Deep Learning for Lung Disease Detection on Medical Images: State-of-the-Art, Taxonomy, Issues and Future Directions AUTHORS: Stefanus Tao Hwa Kieu and Abdullah Bade In this paper, the authors mainly concentrated on the lung disease and survey about deep learning for lung disease detection based on Xray images. The authors mainly studies one survey paper published in the last five years regarding deep learning directed at lung diseases detection. In that paper they saw a taxonomy and analysis of the trend of recent work, but they saw some missing information present in those contents. So in order to improve the information related to this taxonomy they studies nearly 98 articles published from 2016 to 2020.From all these articles they gathered seven main attributes which are required for the lung disease detection such as: image types, features, data augmentation, types of deep learning algorithms, transfer learning, the ensemble of classifiers and types of lung diseases.

### 2.1 Problem Statement

### Recently a large dataset of X-ray lung data was public on Kaggle followed by labeled lung disease data. This is a good condition for us to implement this project. In this project we will conduct a study and analysis of this data set, and then apply Deep Learning to predict that the patient has a lung disease. This project is a binary classification with input is patient's data (X-ray images) and output is found for diseases or not. The difficulty is a new dataset, and we will be one of the pioneers to learn it, our analysis is that this is a large dataset but has never been processed full, data has a lot of noise, and X-ray of the lung is not likely to provide enough information to assess whether a patient may be ill. We will use Deep Learning to process data as well as create models for diagnosing patients. My key point here will be: combining the processing of patient information with data from X-rays, using VGG16 with the well-known pre-trained model.

### 2.2 Existing System and its limitations

### In the existing system, there was no concept like lung cancer prediction using CNN models. All the prediction is done using manual approach or by using primitive Machine Learning models. In the ML we can able to classify whether lung cancer is present or not, but those models cannot classify the records with accuracy and parameters.

### Limitations of existing system

### All the existing schemes are limited to the few classes’ classification only

### All the existing systems are failed to classify the chest x-ray images and then try to find out the cancer symptoms.

### All the existing ML approaches try to classify the patient’s information from the raw dataset

### There is no accurate model to classify the real time chest x-ray for detecting and prediction of accuracy of that image.

### 2.3 Proposed System

### In this proposed work we try to design an application which can be used for prediction of lung cancer from real world chest x-ray images. For training the system we try to collect the sample chest X-ray images which contain cancer symptoms from KAGGLE website and then train the system. Once the system is trained now we can check the model performance by giving dynamic images and check the performance of Model. Using Deep Learning to predict lung diseases from Chest X-rays can be a lifesaving factor for an individual suffering from the disease. This is possible as the results can be predicted with a high percentage of accuracy instantly. This paper presents an effective way for expert diagnosis of lung diseases using Deep Learning. It focuses on creating a system for assistance of Radiologists in detection of lung diseases. This will especially benefit rural areas where radiologists aren’t easily available. We use two model like Vgg16 for predicting the lung cancer from chest X ray images and then tell that model accuracy and performance.

### 2.4 System Requirements

HARDWARE:

* Processor-i3
* Hard disk-8GB
* Memory-1 TB

SOFTWARE:

* Google collabs
* Python programming language
* VGG 16 and VGG 19( Transfer Learning)

# 

# 

# 

# *Chapter 3*

***DESIGN***

## 3. DESIGN

## 3.1 Model Architecture

VGGNet-16 consists of 16 convolutional layers and is very appealing because of its very uniform Architecture. Similar to AlexNet, it has only 3x3 convolutions, but lots of filters. It can be trained on 4 GPUs for 2–3 weeks. It is currently the most preferred choice in thecommunity for extracting features from images. The weight configuration of the VGGNet is publicly available and has been used in many other applications and challenges as a baseline feature extractor.

However, VGGNet consists of 138 million parameters, which can be a bit challenging to handle. VGG can be achieved through transfer Learning. In which the model is pretrained on a dataset and the parameters are updated for better accuracy and you can use the parameters values.

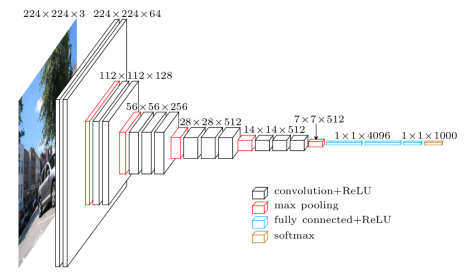


Figure 2: VGG16 Model Architecture

The input to cov1 layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, centre). In one of the configurations, it also utilizes 1×1 convolution filters, which can be seen as a linear transformation of the input channels (followed by non-linearity). The convolution stride is fixed to 1 pixel; the spatial padding of conv. layer input is such that the spatial resolution is preserved after convolution, i.e. the padding is 1-pixel for 3×3 conv. layers. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv.  layers (not all the

conv. layers are followed by max-pooling). Max-pooling is performed over a 2×2 pixel window, with stride 2.

Three Fully-Connected (FC) layers follow a stack of convolutional layers (which has a different depth in different architectures): the first two have 4096 channels each, the third performs 1000-way ILSVRC classification and thus contains 1000 channels (one for each class). The final layer is the soft-max layer. The configuration of the fully connected layers is the same in all networks.

All hidden layers are equipped with the rectification (ReLU) non-linearity. It is also noted that none of the networks (except for one) contain Local Response Normalisation (LRN), such normalization does not improve the performance on the ILSVRC dataset, but leads to increased memory consumption and computation time.

## 3.2 Components in the Architecture

## 

## Figure 3: VGG16 Layers

### 16 layers of VGG16

1. Convolution using 64 filters  
2. Convolution using 64 filters + Max pooling  
3. Convolution using 128 filters  
4. Convolution using 128 filters + Max pooling  
5. Convolution using 256 filters  
6. Convolution using 256 filters  
7. Convolution using 256 filters + Max pooling  
8. Convolution using 512 filters  
9. Convolution using 512 filters  
10. Convolution using 512 filters+Max pooling  
11. Convolution using 512 filters  
12. Convolution using 512 filters  
13. Convolution using 512 filters+Max pooling  
14. Fully connected with 4096 nodes  
15. Fully connected with 4096 nodes  
16. Output layer with Softmax activation with 1000 nodes.

# *Chapter 4*

***IMPLEMENTATION***

## 4. IMPLEMENTATION

## 4.1 Methodology Description

## In this section we try to discuss about proposed CNN model which is used to detect chest disease detection using pre-trained CNN models such as VGG16 and VGG19. The Application is mainly divided into 4 modules.

## They are as follows:

## Convolution Layer

## Rectified Linear Unit (RELU) Layer

## 3. Pooling Layer

## 4. Fully Connected layer

#### 4.1.1. Convolutional Layer

#### A convolution is defined as an operation on two functions. In image analysis, one function consists of input values (e.g. pixel values) at a position in the image, and the second function is a filter (or kernel) each can be represented as array of numbers. Computing the dot product between the two functions gives an output. The filter is then shifted to the next position in the image as defined by the stride length. The computation is repeated until the entire image is covered, producing a feature (or activation) map. This is a map of where the filter is stro0ngly activated and ‘sees’ a feature such as a straight line, a dot, or a curved edge. If a photograph of a face was fed into a CNN, initially low-level features such as lines and edges are discovered by the filters. These build up to progressively higher features in subsequent layers, such as a nose, eye or ear, as the feature maps become inputs for the next layer in the CNN architecture.

#### 4.1.2. Rectified Linear Unit Layer

#### The RELU layer is an activation function that sets negative input values to zero. This simplifies and accelerates calculations and training, and helps to avoid the vanishing gradient problem. Mathematically it is defined as: f(x) =max (0, x). Where x is the input to the neuron. Other activation functions include the sigmoid, tanh, leaky RELUs, Randomized RELUs and parametric RELUs.

#### 

#### 4.1.3. Pooling Layer

#### The Pooling layer is inserted between the Convolution and RELU layers to reduce the number of parameters to be calculated, as well as the size of the image (width and height, but not depth). Max-pooling is most commonly used; other pooling layers include Average pooling and L2-normalization pooling. Maxpooling simply takes the largest input value within a filter

#### and discards the other values; effectively it summarizes the strongest activations over a neighbourhood. The rationale is that the relative location of a strongly activated feature to another is more important than its exact location.

#### 4.1.4. Fully Connected Layer

#### The final layer in a CNN is the Fully Connected Layer, meaning that every neuron in the preceding layer is connected to every neuron in the Fully Connected Layer. Like the convolution, RELU and pooling layers, there can be 1 or more fully connected layers depending on the level of feature abstraction desired. This layer takes the output from the preceding layer (Convolution, RELU or Pooling) as its input, and computes a probability score for classification into the different available classes. In essence, this layer looks at the combination of the most strongly activated features that would indicate the image belongs to a particular class. For example, on histology glass slides, cancer cells have a high DNA to cytoplasm ratio compared to normal cells. If features of DNA were strongly detected from the preceding layer, the CNN would be more likely to predict the presence of cancer cells.

# 

# *Chapter 5*

# *RESULTS*

## 5. RESULTS

## 5.1 Step wise Code Process

## Here I first importing all the libraries which i will need to implement VGG16. I will be using Sequential method as I am creating a sequential model. Sequential model means that all the layers of the model will be arranged in sequence. Here I have imported ImageDataGenerator from keras preprocessing. The objective of ImageDataGenerator is to import data with labels easily into the model.

## Then we are resize all the images in the dataset into the parameters as (224,224) that which 224 represents height and 224 represents width.

## Then I have created an object of ImageDataGenerator for both training and testing data.

## Then the data is ready to be passed to the neural network.

## VGG16 model is now implemented with an input layer and dense layer as output and convolutional and max pooling layers in the middle via transfer learning.

## Now the developed model is complied

## Now by using image data generator the images are imported from the data set and categorized into test data and train data.

## Now train the dataset for 5 epochs, which means we are training the model with all the datasets for 5 cycles.

## Now the loss and accuracy are represented in a graph format with accuracy of 97% approx. and loss of 0.06%.

## Now the we need to give images to our model which it will predict that the x ray image is a normal image or diseased one.

## 5.2 Source Code

*# import the libraries as shown below*

**from** keras.layers **import** Input, Lambda, Dense, Flatten

**from** keras.models **import** Model

*#from keras.applications.resnet50 import ResNet50*

**from** keras.applications.vgg16 **import** VGG16

**from** keras.applications.vgg16 **import** preprocess\_input

**from** keras.preprocessing **import** image

**from** keras.preprocessing.image **import** ImageDataGenerator

**from** keras.models **import** Sequential

**import** numpy **as** np

**from** glob **import** glob

**import** matplotlib.pyplot **as** plt

# re-size all the images to this

train\_path = '/content/drive/MyDrive/chest\_xray/train'

valid\_path = '/content/drive/MyDrive/chest\_xray/val'

IMAGE\_SIZE = [224, 224]

vgg = VGG16(input\_shape=IMAGE\_SIZE + [3], weights='imagenet', include\_top=False)

from google.colab import drive

drive.mount('/content/drive')

for layer in vgg.layers:

    layer.trainable = False

# useful for getting number of output classes

folders = glob('/content/drive/MyDrive/chest\_xray/train/\*')

# our layers - you can add more if you want

x = Flatten()(vgg.output)

prediction = Dense(len(folders), activation='softmax')(x)

# create a model object

model = Model(inputs=vgg.input, outputs=prediction)

# view the structure of the model

model.summary()

model.compile(

  loss='categorical\_crossentropy',

  optimizer='adam',

  metrics=['accuracy']

)

# Use the Image Data Generator to import the images from the dataset

from keras.preprocessing.image import ImageDataGenerator

train\_datagen = ImageDataGenerator(rescale = 1./255,

                                   shear\_range = 0.2,

                                   zoom\_range = 0.2,

                                   horizontal\_flip = True)

# Make sure you provide the same target size as initialied for the image size

training\_set = train\_datagen.flow\_from\_directory('/content/drive/MyDrive/chest\_xray/train',

                                       target\_size = (224, 224),

                                       batch\_size = 32,

                                       class\_mode = 'categorical')

test\_set = test\_datagen.flow\_from\_directory('/content/drive/MyDrive/chest\_xray/test',

                                       target\_size = (224, 224),

                                       batch\_size = 32,

                                       class\_mode = 'categorical')

# fit the model

# Run the cell. It will take some time to execute

r = model.fit\_generator(

  training\_set,

  validation\_data=test\_set,

  epochs=5,

  steps\_per\_epoch=len(training\_set),

  validation\_steps=len(test\_set)

)

# plot the loss

plt.plot(r.history['loss'], label='train loss')

plt.plot(r.history['val\_loss'], label='val loss')

plt.legend()

plt.show()

plt.savefig('LossVal\_loss')

# plot the accuracy

plt.plot(r.history['accuracy'], label='train acc')

plt.plot(r.history['val\_accuracy'], label='val acc')

plt.legend()

plt.show()

plt.savefig('AccVal\_acc')

import tensorflow as tf

from keras.models import load\_model

model.save('model\_vgg16.h5')

from keras.applications.vgg19 import VGG19

from keras.applications.inception\_v3 import InceptionV3

from keras.models import load\_model

from keras.preprocessing import image

from keras.applications.vgg16 import preprocess\_input

import numpy as np

model= load\_model('/content/model\_vgg16.h5')

img=image.load\_img('/content/drive/MyDrive/chest\_xray/val/PNEUMONIA/person1946\_bacteria\_4874.jpeg', target\_size = (224,224))

x=image.img\_to\_array(img)

x=np.expand\_dims(x,axis=0)

img\_data=preprocess\_input(x)

classes=model.predict(img\_data)

import pandas as pd

df=pd.DataFrame(classes, columns=['Normal','Pneumonia'])

print(df)

## 5.3 Resulting images

## 

## Figure 4: VGG16 layers through transfer learning

## 

## Figure 5: Training loss data Graph

## 

## Figure 6: Training validation accuracy graph

## 

## Figure 7: Prediction of Disease

# 

# 

# 

# 

# *Chapter 6*

***CONCLUSION***

## 6. CONCLUSION

## 6.1 Conclusion

## In this current work we for the first time designed and implemented an application using deep learning VGG16 a CNN model in the medical field for Chest or Lung disease detection from chest X-ray images. We try to design an application which can able to identify the abnormality present in the human chest or lungs from the affected part of image and then find out the abnormality. At present, it is very interesting to design the deep intricate neural network (CNN) is the latest image recognition solution. Here we try to gather several infected lung images as well as normal lung images and then try to train the system with all these images. Once the model is trained then we try to give a sample chest X-ray image as input and check whether that image is having abnormality or not. To solve the above problem, we developed a deep learning model using a CNN based VGG16 algorithm which is achieved through transfer learning to detect the abnormality present in human chest. By conducting various experiments on our proposed model, we achieved a classification accuracy of 95.0% when applied to the test dataset. If this application is designed perfectly we can able to use this model on covid patients so that we can able to check the current lung status of that covid affected patients and this can greatly reduce lot of physical examination in this pandemic situation.

## 6.2 Issues and Future Direction of Lung Disease Detection Using Deep Learning

## This subsection presents the remaining issues and corresponding future direction of lung disease detection using deep learning, which are the final contributions of this project. The state-of-the-art lung disease detection field is suffering from several issues that can be found in the papers considered. Some of the proposed future works are designed to deal with the issues found. Details of the issues and potential future works are presented in Sections 6.2.1 and 6.2.2, respectively.

#### 6.2.1. Issues

#### This section presents the issues of lung disease detection using deep learning found in the literature. Four main issues were identified: (i) data imbalance; (ii) handling of huge image size; (iii) limited available datasets; and (iv) high correlation of errors when using ensemble techniques.

#### Data imbalance: When doing classification training, if the number of samples of one class is a lot higher than the other class, the resulting model would be biased. It is better to have the same number of images in each class. However, oftentimes that is not the case. For example, when performing a multiclass classification of COVID-19, pneumonia and normal lungs, the number of images for pneumonia far exceeds the number of images for COVID-19.

#### Handling of huge image size: Most researchers reduced the original image size during training to reduce computational cost. It is extremely computationally expensive to train with the original image size, and it is also time-consuming to train a deeply complex model even with the aid of the most powerful GPU hardware.

#### Limited available datasets: Ideally, thousands of images of each class should be obtained for training. This is to produce a more accurate classifier. However, due to the limited number of datasets, the number of available training data is often less than ideal. This causes researchers to search for other alternatives to produce a good classifier.

#### High correlation of errors when using ensemble techniques: It requires a variety of errors for an ensemble of classifiers to perform the best. The base classifiers used should have a very low correlation. This, in turn, will ensure the errors of those classifiers also will be varied. In other words, it is expected that the base classifiers will complement each other to produce better classification results. Most of the studies surveyed only combine classifiers that were trained on similar features. This causes the correlation error of the base classifiers to be high.

#### 6.2.2. Potential Future Works

#### This section presents the possible future works that should be considered to improvise the performance of lung disease detection using deep learning.

#### Make datasets available to the public: Some researchers used private hospital datasets. To obtain larger datasets, efforts such as de-identification of confidential patients’ information can be conducted to make the data public. With more data available, the produced classifiers would be more accurate. This is because, with more data comes more diversity. This decreases the generalization error because the model becomes more general as it was trained on more examples. Medical data are hard to come by. Therefore, if the datasets were made public, more data would be available for researchers.

#### Usage of cloud computing: Performing training using cloud computing might overcome the problem of handling of huge image size. On a local mid-range computer, training with large images will be slow. A high-end computer might speed up the process a little, but it might still be infeasible. However, by training the deep learning model using cloud computing, we can use multiple GPUs at a reasonable cost. This allows higher computational cost training to be conducted faster and cheaper.

#### Usage of more variety of features: Most researchers use features automatically extracted by CNN. Some other features such as SIFT, GIST, Gabor, LBP and HOG were studied. However, many other features are still yet to be explored, for example quadtree and image histogram. Efforts can be directed to studying different types of features. This can address the issue of the high correlation of errors when using ensemble techniques. With more features comes more variation. When combining many variations, the results are often better. Feature engineering allows the extraction of more information from present data. New information is extracted in terms of new features. These features might have a better ability to describe the variance in the training data, thus improving model accuracy.

#### Usage of the ensemble learning: Ensemble techniques show great potentials. Ensemble methods often improve detection accuracy. An ensemble of several features might provide better detection results. An ensemble of different deep learning techniques could also be considered because ensembles perform better if the errors of the base classifiers have a low correlation.

## 

## 

## 7. REFERENCES

## Bousquet, J. Global Surveillance, Prevention and Control of Chronic Respiratory Diseases; World Health Organization: Geneva, Switzerland, 2007; pp. 12–36.

## Forum of International Respiratory Societies. The Global Impact of Respiratory Disease, 2nd ed.; European Respiratory Society, Sheffield, UK, 2017; pp. 5–42.

## World Health Organization. Coronavirus Disease 2019 (COVID-19) Situation Report; Technical Report March; World Health Organization: Geneva, Switzerland, 2020.

## Rahaman, M.M.; Li, C.; Yao, Y.; Kulwa, F.; Rahman, M.A.; Wang, Q.; Qi, S.; Kong, F.; Zhu, X.; Zhao, X. Identification of COVID-19 samples from chest X-Ray images using deep learning: A comparison of transfer learning approaches. J. X-Ray Sci. Technol. 2020, 28, 821–839. [CrossRef]

## Yahiaoui, A.; Er, O.; Yumusak, N. A new method of automatic recognition for tuberculosis disease diagnosis using support vector machines. Biomed. Res. 2017, 28, 4208–4212.