PNEUMONIA DETECTION USING DEEP CONVOLUTIONA NEURAL NETWORKS

Submitted in partial fulfillment of the requirements for the award of degree of

BACHELOR OF ENGINEERING
IN
INFORMATION TECHNOLOGY

Submitted to:
NAVPREET KAUR

Submitted By:
MEHAK SURYAVANSHY
SHREYA LINGWAL
UID
17BET1063
17BET1053

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Chandigarh University, Gharuan

APRIL 2020

CERTIFICATE

We students of 'Bachelor Of Engineering in Information Technology', session:2017-2021,Chandigarh University, Punjab, hereby declare that the work presented in this project work entitled 'PNEUMONIA DETECTION USING DEEP CONVOLUTIONAL NEURAL NETWORK'. It is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning ,except where due acknowledgement has been made in the text.

MEHAK SURYAVANSHY
SHREYA LINGWAL
UID: 17BET1063
17BET1053

Date: 13/04/2020

Place: Chandigarh University

DECLARATION

We , student of **Bachelor of Engineering** in **Computer Science & Engineering, 6th Semester** , **session: Jan–May 2020, Chandigarh University,** hereby declare that the work presented in this Project Report entitled **" Pnuemonia Detection"** is the outcome of my own work, is bona fide and correct to the best of my knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any other university or anywhere else for the award of any degree or any professional diploma.

Student details and Signature

APPROVED & GUIDED BY:

ACKNOWLEDGEMENT

This project was a great chance for learning and professional development. Therefore, we consider ourself as a very lucky team as we were provided with an opportunity to be a part of it. We are also grateful for having a chance to meet so many wonderful people and professionals.

Bearing in mind previous we are using this opportunity to express our deepest gratitude and special thanks to NAVPREET KAUR who in spite of being extraordinarily busy with her duties ,took time to hear, guide and keep us on the correct path and allowing us to carry out our project.

We express our deepest thanks for taking part in useful decision & giving necessary advices and guidance and arranged all facilities to make life easier. We choose this moment to acknowledge her contribution gratefully.

It is our radiant sentiment to place on record my best regards, deepest sense of gratitude and for her careful and precious guidance which were extremely valuable for our study both theoretically and practically .

we perceive as this opportunity as a big milestone in my career development. We will strive to use gained skills and knowledge in the best possible way, and we will continue to work on their improvement, in order to attain desired career objectives.

Last but not the least my project members who shared valuable information that helped in the successful completion of the project.

MEHAK SURYAVANSHY

SHREYA LINGWAL

Table of Contents

LIST OF FIGURES

ABSTRACT

The risk of pneumonia is immense for many, especially in developing nations where billions face energy poverty and rely on polluting forms of energy. The WHO estimates that over 4 million premature deaths occur annually from household air pollution-related diseases including pneumonia. Over 150 million people get infected with pneumonia on an annual basis especially children under 5 years old. In such regions, the problem can be further aggravated due to the dearth of medical resources and personnel. For example, in Africa's 57 nations, a gap of 2.3 million doctors and nurses exists. For these populations, accurate and fast diagnosis means everything. It can guarantee timely access to treatment and save much needed time and money for those already experiencing poverty.

There is a great growing interest in the domain of deep learning techniques for identifying and classifying images with various datasets. An enormous availability of datasets has developed a keen interest in deep learning. Pneumonia is a disease that is caused by various bacteria, virus etc. X-ray is one of the major diagnosis tools for diagnosing pneumonia. This work mainly proposes a convolutional neural system (CNN) model prepared without any preparation to group and identify the occurrence of pneumonia disease from a given assortment of chest X-ray image tests. Dissimilar to different strategies that depend exclusively on more learning draws near or conventional carefully assembled systems to accomplish an amazing grouping execution, and developed a convolutional neural arrange model without any preparation to separate and character the images to decide whether an individual is suffering with pneumonia. This model could help alleviate the dependability and difficult challenges frequently confronted to manage therapeutic problems. In this paper, CNN algorithm has been used along with different data augmentation techniques for improving the classification accuracies which has been discussed to increase the performance which will help in improving the validation and training accuracies and characterization of exactness of the CNN model and accomplished various results.

REFERENCES

[1] https://towardsdatascience.com/deep-learning-for-detecting-pneumonia-from-x-ray-images-fc9a3d9fdba8

[2] http://www.ijitee.org/wp-content/uploads/papers/v9i4/D1599029420.pdf

[3] https://www.hindawi.com/journals/jhe/2019/4180949/

[4] http://www.ijstr.org/final-print/oct2019/Deep-Learning-Approach-For-Prediction-Of-Pneumonia.pdf

[5] https://s3.amazonaws.com/academia.edu.documents/62430641/B22010408112020032196653-12qkcu9.pdf?response-content-disposition=inline%3B%20filename%3DPneumonia_Detection_through_X-Ray_Using.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-

CHAPTER 1

INTRODUCTION

The risk of pneumonia is immense for many, especially in developing nations where billions face energy poverty and rely on polluting forms of energy. The WHO estimates that over 4 million premature deaths occur annually from household air pollution-related diseases including pneumonia. Over 150 million people get infected with pneumonia on an annual basis especially children under 5 years old. In such regions, the problem can be further aggravated due to the dearth of medical resources and personnel. For example, in Africa’s 57 nations, a gap of 2.3 million doctors and nurses exists. For these populations, accurate and fast diagnosis means everything. It can guarantee timely access to treatment and save much needed time and money for those already experiencing poverty.

Deep neural network models have conventionally been designed, and experiments were performed upon them by human experts in a continuing trial-and-error method. This process demands enormous time, know-how, and resources. To overcome this problem, a novel but simple model is introduced to automatically perform optimal classification tasks with deep neural network architecture. The neural network architecture was specifically designed for pneumonia image classification tasks. The proposed technique is based on the convolutional neural network algorithm, utilizing a set of neurons to convolve on a given image and extract relevant features from them. Demonstration of the efficacy of the proposed method with the minimization of the computational cost as the focal point was conducted and compared with the exiting state-of-the-art pneumonia classification networks.

In recent times, CNN-motivated deep learning algorithms have become the standard choice for medical image classifications although the state-of-the-art CNN-based classification techniques pose similar fixated network architectures of the trial-and-error system which have been their designing principle. U-Net, SegNet, and CardiacNet are some of the prominent architectures for medical image examination. To design these models, specialists often have a large number of choices to make design decisions, and intuition significantly guides manual search process. Models like evolutionary-based algorithms and reinforcement learning (RL) have been introduced to locate optimum network hyperparameters during training. However, these techniques are computationally expensive, gulping a ton of processing power. As an alternative, our study proposes a conceptually simple yet efficient network model to handle the pneumonia classification problem as shown in Figures 1 and 2.

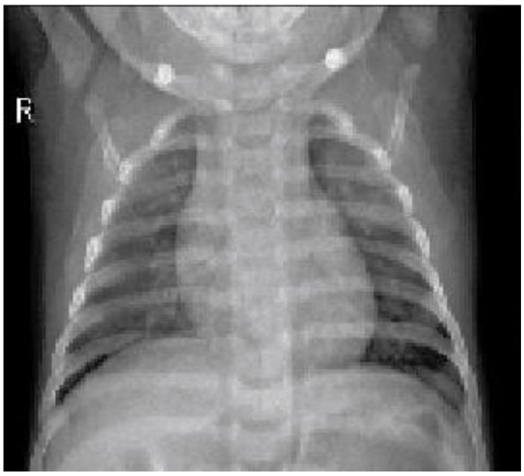


Fig. 1.1 Image without Pneumonia

Fig. 1.2 Image with Pneumonia

CNNs have an edge over DNNs by possessing a visual processing scheme that is equivalent to that of humans and extremely optimized structure for handling images and 2D and 3D shapes, as well as ability to extract abstract 2D features through learning. The max-pooling layer of the convolutional neural network is effective in variant shape absorptions and comprises sparse connections in conjunction with tied weights. When compared with fully connected (FC) networks of equivalent size, CNNs have a considerably smaller amount of parameters. Most importantly, gradient-based learning algorithms are employed in training CNNs and they are less prone to diminishing gradient problem. Since the gradient-based algorithm is responsible for training the whole network in order to directly diminish an error criterion, highly optimized weights can be produced by CNNs.

Deep Learning (DL) methods are vanquishing over the predominant customary methodologies of neural system, with regards to the tremendous measure of dataset, applications requiring complex capacities requesting increment precision with lower time complexities. It is a fact that the disease like pneumonia is spreading very vast and also its threat is very tremendous and causing a barrier in developing a disease free nation. It has been predicted by WHO that 4 million sudden misfortunes happen each year from nuclear family air tainting diseases, maximum people are suffering from pneumonia disease. Also, it has been found in a survey that approx. 160 million people were suffered from pneumonia in which there were children of under 5 years of age. In such territories, the issue can be moreover bothered on account of the insufficiency o helpful resources and staff.

A survey on Africa has depicted about various nations which were effected from pneumonia. For different types of people, this technique proves to be one of the best techniques which imply the best treatment of the pneumonia disease with effective results. Previously, various models and architectures were designed for the evaluation of such kind of diseases. Various types of experimental strategies were attempted in order to get the best results out of the all other methods. But the techniques used for detecting and diagnosing such types of diseases are time consuming. The deep learning techniques are mainly part of the artificial neural networks which acquires great power and versatility for the learning process. To avoid all kinds of issues, a new anyway essential model is familiar with thusly perform perfect gathering endeavours with significant neural framework building. The neural network building was unequivocally expected for pneumonia diagnosis and then classifying the images. The proposed methodology relies upon the convolutional neural framework computation, utilizing various numbers of neurons for convolving on a given picture and concentrate appropriate features from them. This paper presents great feasibility in several aspects as the purpose of combination was coordinated and differentiated and also for improving the pneumonia diagnosing frameworks. Starting late, CNN energized significant learning estimations have gotten the standard choice for remedial picture orders in spite of the way that the top tier CNN-based course of action methodology present equivalent centered framework structures of the experimentation system. U-Net, Seg-Net, and Cardiac-Net are a segment of the indisputable plans for remedial picture evaluation. There were several models used as formative based computations and bolster learning (RL) have been familiar with discover perfect framework hyper parameters during getting ready. In any case, these methodologies are very computational, which includes tremendous measure of planning. Using another option, our assessment introduced a hypothetically essential and beneficial framework model to manage the difficulties associated with pneumonia request issue. CNNs are usually more preferred over DNNs by having a visual taking care of plan which is equal and incredibly improved structure for managing pictures and 2D and 3D objects, similarly as ability for isolating dynamic 2D incorporates by using various learning approaches. The CNN contains the huge pooling layer framework which is convincing alive and well digestions and contains pitiful affiliations identified with tied burdens.

CHAPTER 2

SRS

2.1 Purpose

A Software Requirements Specifications (SRS) is a document that describes the nature of the project. In simple words, SRS document is a manual of a project provided it is prepared before you start a project. The purpose of this document is to build an online mail system using Python.

Moreover the use of python language and its libraries make it more simple as they have pre defined functions which are only to be called according to the need of the project.

2.2 FEASIBILITY STUDY

Convolutional Neural Networks (CNNs) consists of various types of layers along with the max pooling layer. It also contains the RELU called as Rectified Linear Unit which helps in ensuring for the non-linearity of the network model. They are not much different from the ANNs. CNN is one of the most popular deep learning neural networks. The first time when CNN came into existence was 2012 when AlexNet was introduced with just 8 layers. Further, it was improved to 152 layers. CNN is mostly used for all the image related problems. One of the most important reason for using the CNN technique was that it helps in the automatic detection of important features without any human involvement. CNN technique is very effective and computationally efficient as it uses various layers and helps in parameter sharing. The subsequent explanation is halting or decreasing the impacts of over fitting. Over fitting is fundamentally when a system can't adapt successfully because of a number of reasons. It is a significant idea of most, if not all AI algorithms and it is significant that each precautionary measure is taken as to lessen its effects. A model should be designed using such considerations that it should reduce the problems of overfitting and underfitting and also it should have the generalizability property for being in the best model category. Using minimum parameters, there will be less chances of the model that it will undergo in overfitting problems and eventually it will help in improving the overall prescient presentation of the architectural model.

2.2.1 FUNCTIONS OF CNN LAYERS

1.Convolutional Layer- This layer mainly focuses on how the CNNs operate and works. The major parameters of this layer mainly focus on the use of learnable kernels. The kernels here are small in dimensionality. Whenever the data hits any convolutional layer, the layer convolves each filter across the spatial dimensionality of the input and produces a 2D activation map. These maps contain the pixel values of the image. The convolutional layers can easily reduce the complexity of any model by optimizing the output produced. This process of optimization can be done using three main hyper parameters, depth, stride and zero padding. By using these, three parameters, we can easily reduce the size and dimensionality of the parameters of convolutional layers output.

2.Pooling Layer- The layer which is mainly responsible for lessening the overall features and dimensionality of the represented image and further, this layer performs various operations which results in lessening the complexity and parameters of the input images. It performs over activation map of the input and then scales out the dimensionality by using the MAX function. The max pooling layer helps in reducing the activation map around 25% from its original size but maintains its depth volume. There are two types of main pooling used in CNN architecture as overlapping pooling and general pooling.

3.Fully Connected Layer- In this, each neuron o every layer is connected with previous layer's neurons. This layer helps in producing the output from the extracted features and then forwarding it to the output layer.

2.2.2 METHODS USED IN THE MODEL

A. Problem Setting : The problem statement for this classification problem mainly consists of chest X-rays dataset and classifying the images with the help of various data augmentation techniques. There are different images which belong to various classes and it becomes very difficult to classify those images correctly on the basis of their features and properties. Also, the main problems to characterize the features of the images and them classify them with improved accuracy and also having less loss of data. Therefore, in the classification process, it is very much essential that the data should not loss. Otherwise, it becomes very difficult to classify the images correctly.

B. Datasets : The dataset consists of main three folders that is training, testing and validations folders having a total images 5836 in numbers. Further, these folders are subdivided into two subfolders as pneumonia and normal folders. The Data Augmentation techniques have helped to perform various types of operations in the images. Images are having images of anterior and posterior chests and they are precisely chosen from retrospective pediatric patients is in between 1 to 6 years. This experiment was conducted to improve the validation accuracy and minimizing the validation loss. The main goal is to obtain the classified images of pneumonia patients using this chest X-ray dataset. In order to maintain the proportion of several data, the original dataset having training and validation sets is modified. Therefore, the training and validation data has been rearranged. There are total of 3628 images that were allocated to the training set and 2208 images allocated to validation set. This modification has helped to improve the validation accuracy to a great extent.

C. Preprocessing : This process mainly involves the transformation of the raw data before it is fed to the deep learning algorithm. When the dataset is collected from various resources, it is gathered in raw format. At this stage, the raw image is not feasible for the analysis and therefore there is a need of preprocessing. It is the way toward changing every data test from multiple points of view and including the entirety of the enlarged examples to the dataset. By doing this one can expand the successful size of the dataset. Changes to apply are generally picked arbitrarily from the predefined set.

D. Data Augmentation : This technique is basically used to enable the researchers for enhancing the data diversity for various data models. Data augmentation techniques involves cropping of data, shifting the data, rotating, padding, flipping etc. and these techniques are used to train the neural networks. In the proposed approach, various data augmentation techniques are being used. The first operation used here is Rescale of 1/255. Then, the next operation is Rotation of 45 degree of images. Further width shift and height shifts of 0.2 is used. Further, the operations used are shear range, zoom range, and horizontal flip. At the point when we feed picture information into a neural system, there are a few highlights of the pictures that we might want the neural system to consolidate or abridge into a lot of numbers or loads. On account of picture characterization, these highlights or flag are the pixels which make up the item in the image. Then again, there are highlights of the pictures that we dislike the neural system to consolidate in its outline of the pictures (the synopsis is the arrangement of loads). On account of picture characterization, these highlights or clamor are the pixels which structure the foundation in the image. Data augmentation will help to perform various operations on the data for enhancing the classification accuracy of the images. This technique helps to improve and add some effective knowledge about the data for better results. Shapes of the images are altered, flipped, changed for getting the proper knowledge of the images. Augmentation also helps in creating the multiple versions of the same images which will in turn enhance the size of the training set. This will help in generalizing the data by improving the efficiency if the training dataset.

CHAPTER 3
ARCHITECTURE DESIGN

3.1 CNN ARCHITECTURE

CNNs basically center on the premise that the info will be included pictures. Such architectures would help in managing different types of data using various datasets. Fig 1 shows the flow diagram of all the layers that how each process works step by step. The major key contrasts is that the neurons present inside the CNN model are involved neurons composed into three measurements, the spatial dimensionality of the info (stature and the width) and the profundity. The profundity doesn't allude to the all out number of layers inside the ANN, yet the third element of an initiation volume. Not at all like standard ANNS, the neurons inside some random layer will just associate with a little area of the layer going before it. CNNs are contained three sorts of layers. These are convolutional layers pooling layers and completely associated layers. At the point when these layers are stacked, a CNN technique has been framed. The working of the CNN model has been categorized into four main functions as given below:

1. Firstly, there is an input layer which is used for holding the pixel values of the image.
2. Then, the convolution layer is there which helps in determining the output of several neurons and these neurons are being connected to the local regions. Then, the further calculation is being done by scalar product between their weights and with the regions which is connected to the input volume. After this the Rectified Linear Unit (ReLu) is there which has a function of applying an activation function which is done element wise like sigmoid function to the output which is produced by the activation of the previous layer.
3. Then, the pooling layer is there which is used to down sample the spatial dimensionality of the input and then it reduces the various parameters and shorten the image sometimes to its half within that activation.
4. The fully connected layers help in producing the various scores obtained from the

activations. The main aim of this layer is that it takes the results from the convolution or pooling layer and then us that result to classify the image into a form of label. After this they pass the obtained result to the output layer, where each neuron will represent a classification label.

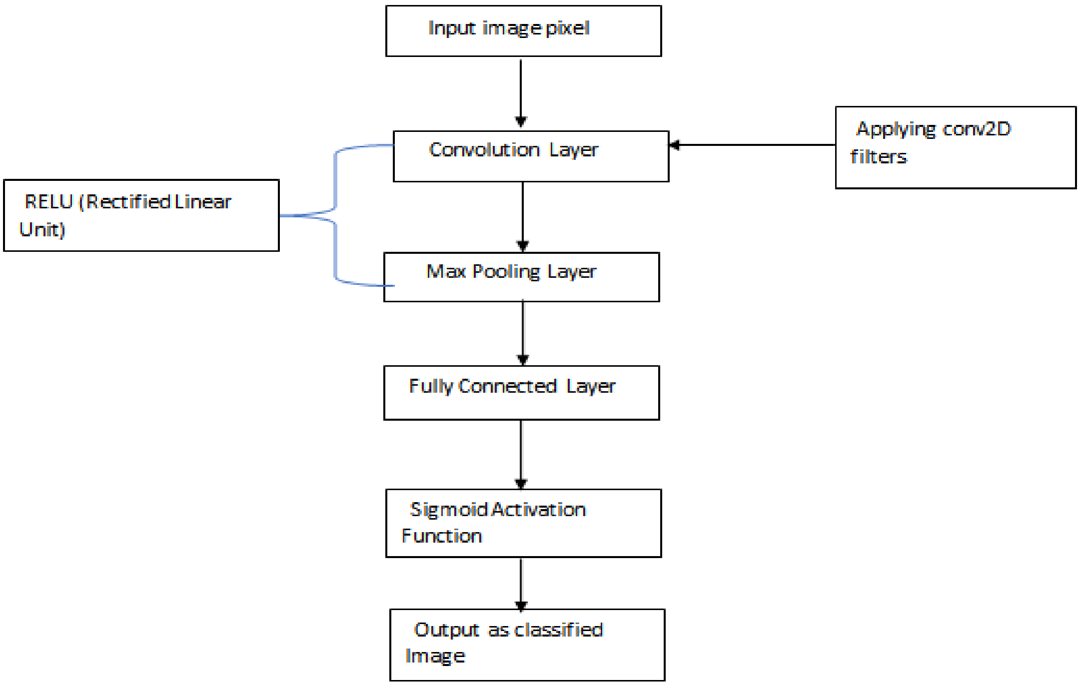


Fig.3.1 Flowchart of CNN model

3.1.1 FUNCTIONS OF CNN LAYERS

1. Convolutional Layer- This layer mainly focuses on how the CNNs operate and works. The major parameters of this layer mainly focus on the use of learnable kernels. The kernels here are small in dimensionality. Whenever the data hits any convolutional layer, the layer convolves each filter across the spatial dimensionality of the input and produces a 2D activation map. These maps contain the pixel values of the image. The convolutional layers can easily reduce the complexity of any model by optimizing the output produced. This process of optimization can be done using three main hyper parameters, depth, stride and zero padding. By using these, three parameters, we can easily reduce the size and dimensionality of the parameters of convolutional layers output.

2. Pooling Layer- The layer which is mainly responsible for lessening the overall features and dimensionality of the represented image and further, this layer performs various operations which results in lessening the complexity and parameters of the input images. It performs over activation map of the input and then scales out the dimensionality by using the MAX function.

The max pooling layer helps in reducing the activation map around 25% from its original size but maintains its depth volume. There are two types of main pooling used in CNN architecture as overlapping pooling and general pooling.

3. **Fully Connected Layer-** In this, each neuron o every layer is connected with previous layer's neurons. This layer helps in producing the output from the extracted features and then forwarding it to the output layer.

3.2 MODEL

Here, Fig 3.2 basically represents the complete architecture of the CNN model which is merely divided into several layers. These layers are referred to as the dense layers. It is also having a classifier called as Sigmoid Activation Function. The output of each layer is being forwarded in the next proceeding layer as its input in all the feature extraction layers. The proposed CNN architecture is having combination of several layers like Convolution layers, max pooling and various classification layers.

The layers for feature extractors consists of conv3×3, 32, conv3×3, 32, conv3×3, 64, conv3×3, 128, conv3×3, 128, conv3×3, 128 and RELU activators in between them. Then, the output obtained from the convolutional layers and max pooling layers are being converted into 2D planes which are called as feature maps and further we get the feature maps, respectively for the convolution operations and pooling operations. The size of input image is 200×200×3. Moreover, the plane of each layer was obtained by merging more planes occurred in the previous layers.

In this model, Sigmoid Activation Function is used as the classifier which is kept at far end of the model. It is having a lot of dense layers so it is also called as ANN model. This function is sometimes also called as the squashing function. They limit the output range in between 0 and 1, which helps in the possible prediction of probabilities.

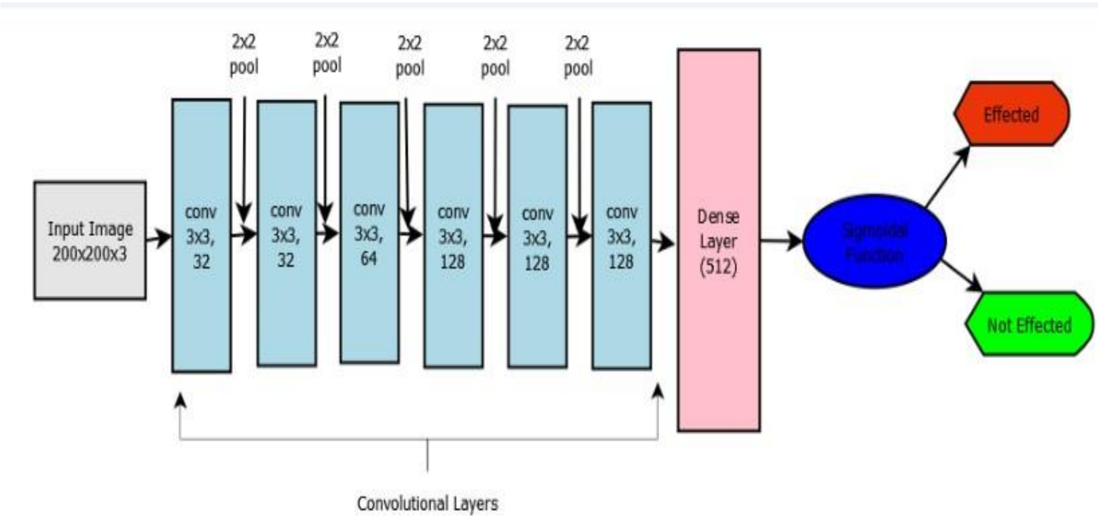


Fig. 3.2 CNN Architecture

The classifier used in this model also requires individual features for each computation involved in the classification process. Hence, the output obtained after the process of feature extraction is again converted into the 1 D feature extractor planes and this mechanism is called the flattening process in which the output obtained is generalized into a lengthy feature vector plane so that it can be utilized in the last classification process. Further, the classification layer is having the flattening layer, dense layer, dropout of 0.5, RELU activator and a sigmoid function used for the final classification of the images. Further, dropout feature is being used of size 0.5, and then layers of size 512 and 1 are being used. Further, in the proposed method, RELU function has been used along with the sigmoid activation function for classifying the images into positive and negative pneumonia. CNN architecture used with the data augmentation techniques has helped in achieving greater accuracy and reduced the validation loss to a greater extent.

CHAPTER 4

PROJECT METHODOLOGY

The proposed model aims to achieve maximum accuracy for pneumonia detection using techniques of data augmentation, residual networks, and stochastic gradient with restarts, cosine annealing and differential learning rates. Fig 4.1 shows the process flow of model development. Further sub- sections give detailed information about the process of the proposed work.

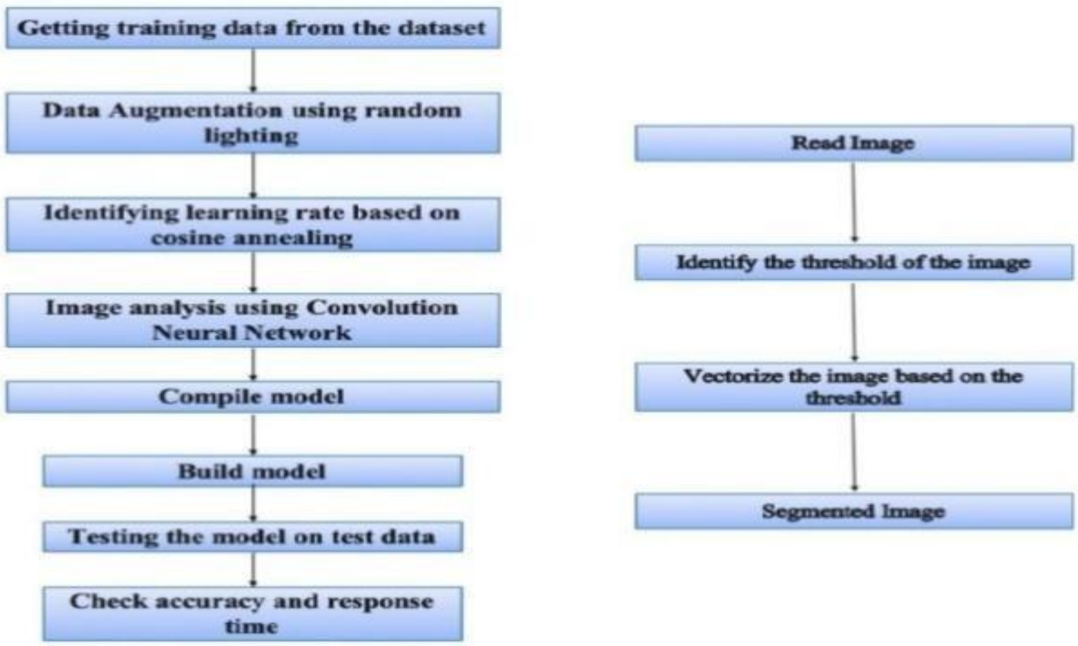


Fig.4.1. Process Flow Of Model Development

A. Dataset: The Kaggle dataset is used consists of total, 5863 X-Ray images (JPEG) . The dataset structured into 3 parts such as train, test, and validation, contains sub folders for each image category or class such as Pneumonia or Normal. Chest X-ray images (anterior-posterior) were looked over review accomplices of pediatric patients of 1 to 5 years from Guangzhou Women and Children's Medical Center, Guangzhou. All chest X-ray imaging was gathered and executed as a major aspect of patients' normal clinical consideration.

B. Data Augmentation: In AI when model is prepared, it tunes its parameters with the end goal that it can outline specific info (an image) to some yield (a class label).As number of parameters increase, a proportional amount of examples need to be shown to the model for achieving good performance. A poorly trained neural network may not be able to predict satisfactory results if the images (x-rays) in the target application change color, brightness act. As the dataset is quite less, and our target application may exist in a variety of conditions, data augmentation is used to create new data with different orientations. Data augmentation was done in order to generate more data and avoid the problem of overfitting. In order to achieve this, random lighting transformation capabilities were used. Using a balanceof0andcontrastof10 data were augmented. The result of augmentation of some images is shown in Fig.4.2.

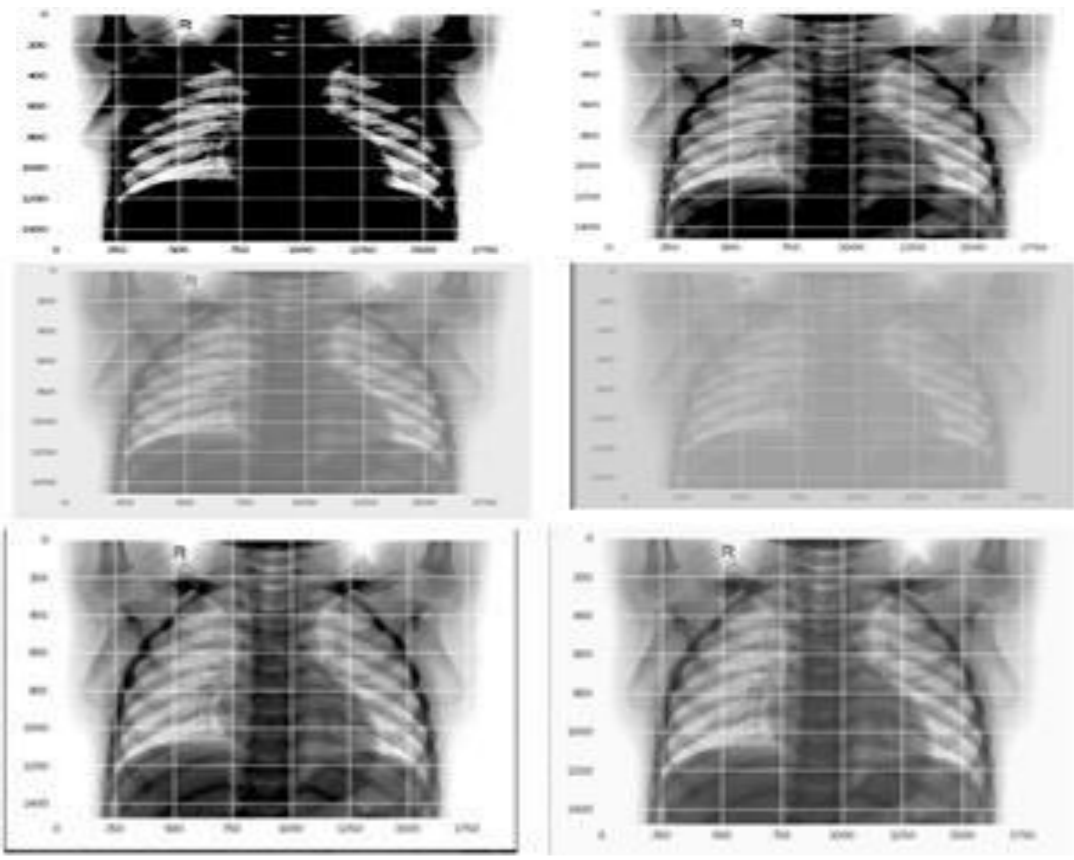


Fig. 4.2. Data augmentation using random lighting with balance 0 and contrast 10

***C.* Analysis using neural networks :** CNNs (Convolutional Neural Networks) are instrumental for performance in image analysis due to convolutional units and, it's ability to include various hidden layers handling convolution and subsampling in order to pull out low to high levels of structures of the input data and hence were used for building the model architecture. A dropout of 0.6 to reduce overfitting. The model was first trained with smaller 64 images using CNN and then steadily increase in image size is done for better CNN and then gradually an increase in the image size were done for better efficiency. Only the last fully connected layer added on top of ResNet34 is trained first.

D. Learning rate with cosine annealing and stochastic gradient with restarts(SGDR): In typical Gradient Descent improvement, like Batch Gradient Descent, the batch is taken to be the entire dataset. Despite the fact that, utilizing the entire dataset is extremely helpful for getting to the minima in a less uproarious or less arbitrary way, however the issue emerges when the datasets get extremely tremendous. In case of Gradient Descent the entire set is required for completing one iteration and it has to be done for each iteration until minima is reached. Due to this, it becomes computationally very expensive. In order to solve this problem Stochastic Gradient Descent is used. Inthis, ituses a single sample rather than the whole dataset for each iteration. The sample is randomly shuffled and chosen for doing the iteration.

For choosing idle learning rate, the graph of loss versus learning was plotted.The estimation of the learning rate was picked dependent on where it is most noteworthy and the misfortune is as yet sliding. As shown in Fig. 4.2 the optimum learning rate is 0.01.As the network gets closer to a global minimum value of loss with each batch of stochastic gradient descent, the learning rate ought to likewise decrease so the calculation does not overshoot.Cosine tempering , a method of decreasing functions admirably with the learning rate, coming about extraordinary outcomes in a computationally proficient way.

During training the model might get stuck at local minima instead of approaching to the global minima. To avoid this, stochastic gradient descent with restarts (SGDR) was incorporated into the model. This adds to the performance of the model in a manner that by rising the learning rate all of sudden, gradient descent may ―hop‖ out of the local minima and find its way near the global minimum. To handle the problem of the model getting stuck at local minima, the learning rate is retune at the start of each epoch to the original value entered as a parameter, and then reduce again over the epoch as described cosine annealing. Further to achieve better accuracy, differential learning rates were introduced. As per the paper, an ideal learning rate can be evaluated via preparing the model at first with a low learning rate and increase its value at each step. When learning rate is too small, loss does not change much, but as learning rate goes higher, loss should decrease faster and faster until a point where it does not decrease anymore and eventually starts increasing.

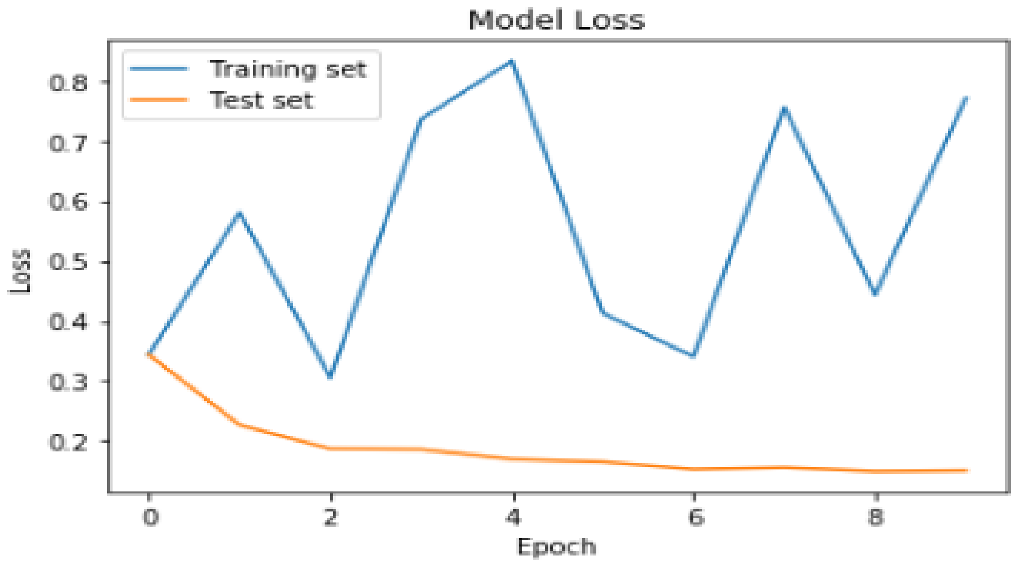


Fig. 4.3 Graph of learning rate versus loss to find optimum learning rate

CHAPTER 5

RESULTS

5.1 SCREENSHOTS

Importing of the required libraries

```
import numpy as np # forlinear algebra
import matplotlib.pyplot as plt #for plotting things
import os
from PIL import Image
print(os.listdir("../input"))

# Keras Libraries
import keras
from keras.models import Sequential
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense
from keras.preprocessing.image import ImageDataGenerator, load_img
```

```
['chest_xray']

Using TensorFlow backend.
```

Our data is located in three folders:

1. train= contains the training data/images for teaching our model.
2. val= contains images which we will use to validate our model. The purpose of this data set is to prevent our model from **Overfitting**. Overfitting is when your model gets a little too comfortable with the training data and can't handle data it hasn't see....too well.
3. test = this contains the data that we use to test the model once it has learned the relationships between the images and their label (Pneumonia/Not-Pneumonia)

```
mainDIR = os.listdir('../input/chest_xray/chest_xray')
print(mainDIR)
```

```
['val', 'train', 'test', '.DS_Store']
```

```
train_folder= '../input/chest_xray/chest_xray/train/'
val_folder = '../input/chest_xray/chest_xray/val/'
test_folder = '../input/chest_xray/chest_xray/test/'
```

Setting up of the training and testing folders.

```
# train
os.listdir(train_folder)
train_n = train_folder+'NORMAL/'
train_p = train_folder+'PNEUMONIA/'
```

```
#Normal pic
print(len(os.listdir(train_n)))
rand_norm= np.random.randint(0,len(os.listdir(train_n)))
norm_pic = os.listdir(train_n)[rand_norm]
print('normal picture title: ',norm_pic)

norm_pic_address = train_n+norm_pic

#Pneumonia
rand_p = np.random.randint(0,len(os.listdir(train_p)))

sic_pic =  os.listdir(train_p)[rand_norm]
sic_address = train_p+sic_pic
print('pneumonia picture title:', sic_pic)

# Load the images
norm_load = Image.open(norm_pic_address)
sic_load = Image.open(sic_address)

#Let's plt these images
f = plt.figure(figsize= (10,6))
a1 = f.add_subplot(1,2,1)
img_plot = plt.imshow(norm_load)
a1.set_title('Normal')

a2 = f.add_subplot(1, 2, 2)
img_plot = plt.imshow(sic_load)
a2.set_title('Pneumonia')
```

```
1342
normal picture title:  NORMAL2-IM-0582-0001.jpeg
pneumonia picture title: person1044_bacteria_2978.jpeg


Text(0.5,1,'Pneumonia')
```

```
# let's build the CNN model

cnn = Sequential()
#add() to add layers to cnn.
#Convolution
cnn.add(Conv2D(32, (3, 3), activation="relu", input_shape=(64, 64, 3)))

#Pooling
cnn.add(MaxPooling2D(pool_size = (2, 2)))

# 2nd Convolution
cnn.add(Conv2D(32, (3, 3), activation="relu"))

# 2nd Pooling layer
cnn.add(MaxPooling2D(pool_size = (2, 2)))
#Flatten serves as a connection between the convolution and dense layers.
# Flatten the layer
cnn.add(Flatten())

# Fully Connected Layers
cnn.add(Dense(activation = 'relu', units = 128))
cnn.add(Dense(activation = 'sigmoid', units = 1))
#The optimizer controls the learning rate.
#The learning rate determines how fast the optimal weights for the model are calculated.
# Compile the Neural network
# 'accuracy' metric to see the accuracy score on the validation set when we train the model.
cnn.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
```

```
# Fitting the CNN to the images
# The function ImageDataGenerator augments your image by iterating through image as your CNN is
#getting ready to process that image

train_datagen = ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range = 0.2,horizont
al_flip = True)

test_datagen = ImageDataGenerator(rescale = 1./255)  #Image normalization.

training_set = train_datagen.flow_from_directory('../input/chest-xray-pneumonia/chest_xray/trai
n/',target_size = (64, 64),batch_size = 32,class_mode = 'binary')

validation_generator = test_datagen.flow_from_directory('../input/chest-xray-pneumonia/chest_xr
ay/val/',target_size=(64, 64),batch_size=32,class_mode='binary')

test_set = test_datagen.flow_from_directory('../input/chest-xray-pneumonia/chest_xray/test/',ta
rget_size = (64, 64), batch_size = 32,class_mode = 'binary')
```

```
Found 5216 images belonging to 2 classes.
Found 16 images belonging to 2 classes.
Found 624 images belonging to 2 classes.
```

```
cnn_model = cnn.fit_generator(training_set,steps_per_epoch = 163,epochs = 10, validation_data =
 validation_generator,validation_steps = 624)
```

```
Epoch 1/10
163/163 [==============================] - 224s 1s/step - loss: 0.3437 - accuracy: 0.8503 -
val_loss: 0.3437 - val_accuracy: 0.8125
Epoch 2/10
163/163 [==============================] - 201s 1s/step - loss: 0.2267 - accuracy: 0.9076 -
val_loss: 0.5804 - val_accuracy: 0.6250
Epoch 3/10
163/163 [==============================] - 198s 1s/step - loss: 0.1871 - accuracy: 0.9273 -
val_loss: 0.3054 - val_accuracy: 0.9375
Epoch 4/10
163/163 [==============================] - 199s 1s/step - loss: 0.1861 - accuracy: 0.9294 -
val_loss: 0.7359 - val_accuracy: 0.6250
Epoch 5/10
163/163 [==============================] - 197s 1s/step - loss: 0.1699 - accuracy: 0.9339 -
val_loss: 0.8328 - val_accuracy: 0.6875
Epoch 6/10
163/163 [==============================] - 199s 1s/step - loss: 0.1655 - accuracy: 0.9360 -
val_loss: 0.4127 - val_accuracy: 0.7500
Epoch 7/10
163/163 [==============================] - 202s 1s/step - loss: 0.1529 - accuracy: 0.9448 -
val_loss: 0.3404 - val_accuracy: 0.8125
Epoch 8/10
163/163 [==============================] - 198s 1s/step - loss: 0.1558 - accuracy: 0.9419 -
val_loss: 0.7556 - val_accuracy: 0.6875
Epoch 9/10
163/163 [==============================] - 198s 1s/step - loss: 0.1498 - accuracy: 0.9444 -
val_loss: 0.4432 - val_accuracy: 0.6875
Epoch 10/10
```

```
test_accu = cnn.evaluate_generator(test_set,steps=624)
acc=test_accu[1]*10
print('The testing accuracy is :', test_accu[1]*100 ,'%')
# Accuracy
plt.plot(cnn_model.history['acc'])
plt.plot(cnn_model.history['val_acc'])
plt.title('Model Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Training set', 'Validation set'], loc='upper left')
plt.show()
```

```
The testing accuracy is : 87.5 %

---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-14-cb58b14180cf> in <module>
      3 print('The testing accuracy is :', test_accu[1]*100 ,'%')
      4 # Accuracy
----> 5 plt.plot(cnn_model.history['acc'])
      6 plt.plot(cnn_model.history['val_acc'])
      7 plt.title('Model Accuracy')

KeyError: 'acc'
```

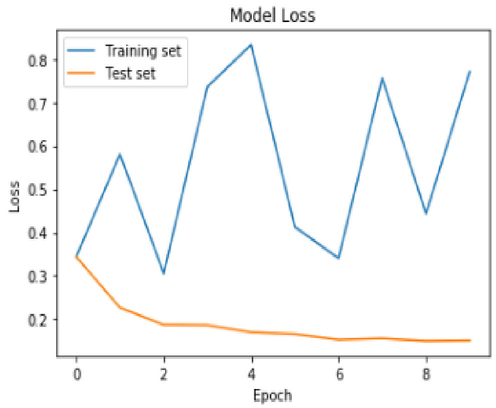
```
# Accuracy
plt.plot(cnn_model.history['acc'])
plt.plot(cnn_model.history['val_acc'])
plt.title('Model Accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Training set', 'Validation set'], loc='upper left')
plt.show()
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-15-fc3210b3bf46> in <module>
      1 # Accuracy
----> 2 plt.plot(cnn_model.history['acc'])
      3 plt.plot(cnn_model.history['val_acc'])
      4 plt.title('Model Accuracy')
      5 plt.ylabel('Accuracy')

KeyError: 'acc'
```

```
# Loss

plt.plot(cnn_model.history['val_loss'])
plt.plot(cnn_model.history['loss'])
plt.title('Model Loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Training set', 'Test set'], loc='upper left')
plt.show()
```



CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

We have demonstrated how to classify positive and negative pneumonia data from a collection of X-ray images. We build our model from scratch, which separates it from other methods that rely heavily on transfer learning approach. In the future, this work will be extended to detect and classify X-ray images consisting of lung cancer and pneumonia. Distinguishing X-ray images that contain lung cancer and pneumonia has been a big issue in recent times, and our next approach will tackle this problem.

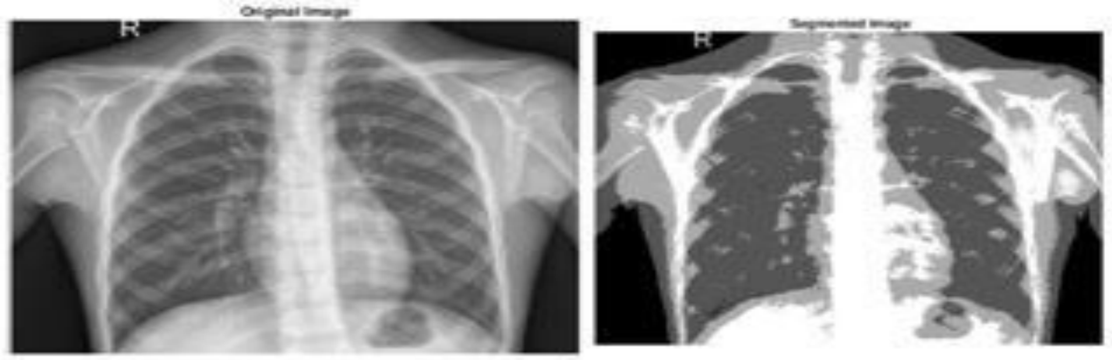


Fig.6.1 Original and segmented image of normal lungs

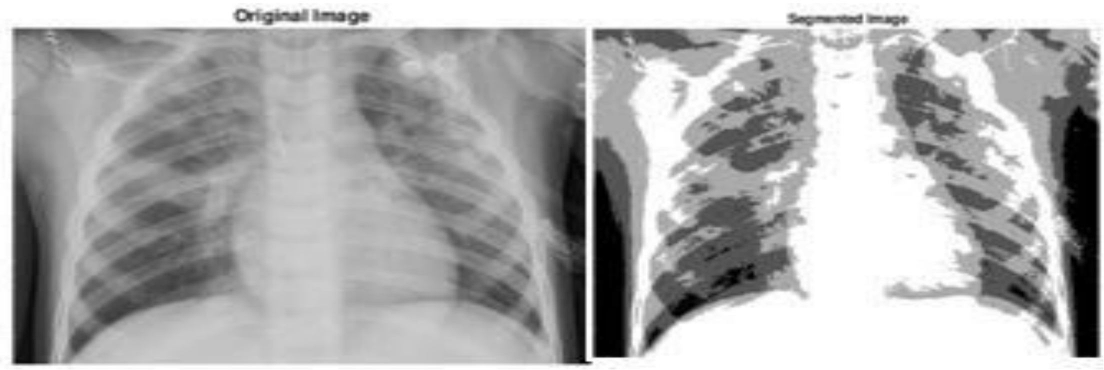


Fig.6.2 Original and segmented image of pneumonia affected lungs

Early diagnosis and treatment of diseases is critical to preventing complications including death. With billions of procedures per year, chest X-ray scans are one of the most common and important diagnosis tools used in practice. They are used for diagnosing and screening a variety of diseases like pneumonia. Statistics by the World

Health Organization indicate that about two thirds of the global population lacks access to proper X-ray diagnostics. Even in the presence of proper imaging equipment the mortality rate is significantly high because of the shortage of experts who can correctly interpret these scans. We develop a public facing platform which is capable of detecting medical anomalies and predictions at levels exceeding medical experts and radiologists. We tested this on the ChestX-ray14 dataset which is one of the largest publicly available chest X-ray dataset. With increasing levels of automation which are simulating the diagnosis done by experts, we hope to achieve an improved healthcare diagnostic access to those parts of the world where there is a shortage of skilled radiologists.

6.2 FUTURE SCOPE

The proposed work will help doctors better predict pneumonia in minimal time with high efficiency. The aggregation of this will contribute to the health care system for better patient satisfaction and care. This work is in its early stages and can be improved by adding more images to the dataset, incorporating better architectures, training the model based on more transformations and orientations.

- CNNs are now-a-days widely used in the computer vision and automation fields. This helps in developing such artificial systems which has capability of performing complex tasks with efficiency.

- CNNs are also being used in the domain of natural language processing for language analysis, language modeling, language designing. CNN models helps in determining the various semantics of any sentence for knowing the better about the client’s requirements.

- CNNs are being used for object detection purpose for identifying the objects in the way. Segmentation of images is also being done using the CNNs.

- Image Classification is one of the very important task which is done using the CNNs in the present scenario by various data augmentation techniques and feature extraction techniques.

- One of the most important applications is the speech recognition in which the speech is being recognized using some automated devices. For example, Google's speech recorder.

- CNNs are also widely used for the data which are computationally very limited in resources. There are several techniques which are still being working on small datasets with improved accuracy of classification.

- CNNs are also being used for the images which are having low resolution. Many researchers have given different techniques to work on the images having low resolution using CNN.