****

**A Project Report and Requirement**

**on**

**Medical Image Analysis Using Deep Learning**

**(CSE 5th Semester Mini project )**

**2021-2022**

**In**

**COMPUTER SCIENCE AND TECHNOLOGY**

**UNDER THE SUPERVISION OF**

**MR.VIJAY SINGH(RESOURCE PERSON)**

***By***

**Nandini Gupta (2014421)**

**Enrollment number : GE -192014421**

**Admission number : 19012744**

**Class roll number : 32**

Department of Computer Science and Engineering

**Graphic Era University, Dehradun**

**Introduction:**

In this project, I have deployed a model that detects whether a person is having Pneumonia or not. The image classification is done by using Convolution Neural Network (CNN). CNN is an artificial neural network that has the ability to detect patterns in the images.

Medical image analysis is a deep learning project in which medical images are analysed using CNN. CNN helps us to classify the images on the basis of certain patterns. In this model I have chosen a dataset having Chest X-Ray images.

**Motivation:**

Pneumonia is an acute respiratory infection that affects the lungs. It is a fatal illness in which the air sacs get filled with pus and other liquid . There are mainly two types of pneumonia: bacterial and viral. Generally, it is observed that bacterial pneumonia causes more acute symptoms. The most significant difference between bacterial and viral pneumonia is the treatment. Treatment of bacterial pneumonia is done using antibiotic therapy, while viral pneumonia will usually get better on its own . It is a prevalent disease all across the globe. Its principal cause includes a high level of pollution. Pneumonia is ranked 8 in the list of the top 10 causes of death in the United States . Due to pneumonia, every year, 3.7 lakh children die in India, which constitutes a total of fifty percent of the pneumonia deaths that occur in India . Children can be protected from pneumonia. It can be prevented with simple interventions and treated with low-cost, low-tech medication and care” .

Therefore, there is an urgent need to do research and development on computer-aided diagnosis so that the pneumonia-related mortality, especially in children, can be reduced.

One of the following tests can be done for pneumonia diagnosis: chest X-rays, CT of the lungs, ultrasound of the chest, needle biopsy of the lung, and MRI of the chest . Currently, chest X-rays are one of the best methods for the detection of pneumonia

**Objective:**

* To capture X-Ray images to analyse the presence of Pneumonia using deep learning techniques.
* To classify X-Ray images as a Pneumonia cases or Normal cases with accuracy.
* To detect the presence of Pneumonia.

**Problem  Statement:**

* In the medical field , Pneumonia is detected by Doctors by referring the X-Ray images which is very time consuming. Therefore , to overcome this problem , an alternative way is to design the system that will automatically identify the presence of Pneumonia in X-Ray images using Convolutional Neural Network and also provide faster and accurate solutions.

**Software requirements:**

Language used: Python 3.7

Operating system: Windows 10

Tool used: Google Collaboratory Notebook

**Hardware Requirements:**

Processor: Intel Core i5 10 Gen

**Methodology followed:**

**Step1: Loading the Dataset**

The dataset that I have used for the image classification is Chest X-Ray images, which consists of 2 categories, Pneumonia and Normal. The data set is categorized into 3 folders (train, test, val) which contains subfolders for each image category Opacity & Normal.

The dataset is extracted directly from the Kaggle using the Kaggle API. For this, I had to create an API token.

Graphical user interface, text, application, email

Description automatically generated

After this, the dataset is downloaded and extracted into a target directory.



Step 2: Initializing the data Now, I needed to import some important libraries and define directory path

Graphical user interface, text, application

Description automatically generated

Step 3: Preparing the data:

a) Data augmentation

Image augmentation technique is used for increasing the size of image training dataset. This is done by flipping, horizontal or vertical shifting, zooming or adding some noises to the same images.

Graphical user interface, text

Description automatically generated

b) Loading the images: There is a class known as flow from directory offered by Image Data Generator which reads the images from folders.

Graphical user interface, text, application

Description automatically generated

Step 4: Applying CNN:

The CNN architecture has convolutional layers which receives inputs and transform the data from the image and pass it as input to the next layer. This transformation is known as the operation of convolutional. We need TensorFlow and necessary libraries for CNN

Text

Description automatically generated

Step 5: Fitting the model

a) Defining callback list I have used Early Stopping which is called to stop the epochs based on some metric and conditions. It helps to avoid overfitting the model. Reduce learning rate when a metric has stopped improving. This callback monitors a quantity and if no improvement is seen for a ‘patience’ number of epochs, the learning rate is reduced

A picture containing chart

Description automatically generated

b) Assigning class weights

Text

Description automatically generated

Step 6: Training the model:

Calendar

Description automatically generated with medium confidence

The EarlyStopping has stopped at 13th epoch at val\_loss =38.8% and val\_accuracy = 75.8%.

Step 7: Evaluating the model:

Text

Description automatically generated with medium confidence

The accuracy of the model is: 90.70%

Confusion matrix:

The upper left (TP) denotes the number of images correctly predicted as normal cases and the bottom right (TN) denotes the correctly predicted number of images as cases of pneumonia. As Pneumonia case, the upper right denotes the number of incorrectly predicted images but were actually normal cases and the lower left denotes the number of incorrectly predicted Normal case images but were actually Pneumonia case.

Graphical user interface, application

Description automatically generated

Classification Report

Precision = True Positives / (True Positives + False Positives)

Recall = True Positives / (True Positives + False Negatives)

F1 = (2 \* Precision \* Recall) / (Precision + Recall)

Graphical user interface

Description automatically generated with low confidence

At last, I have visualized the predicted images using percentages

Graphical user interface, text, application

Description automatically generated

References:

a) For dataset: kaggle.com

b) For resolving errors: https://stackoverflow.com/