International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 08 Issue: 04 | Apr 2021 www.irjet.net p-ISSN: 2395-0072

COVID-19 PNEUMONIA CLASSIFICATION USING DEEP LEARNING TECHNIQUE

RAJASEKARAN G.S[1], SAKTHIVEL GOPI.S[2], VENKATARAMANAN J.S[3]

[1]*Assistant Professor, Department of CSE, Jeppiaar SRR Engineering College, Padur, Cheenai.*
[2, 3]*Final Year Student, Department of CSE, Jeppiaar SRR Engineering College, Padur, Chennai.*

Abstract - The COVID-19 pneumonia symptoms may be similar to other types of viral pneumonia. Because of this, it can be difficult to tell what's causing your condition without being tested for COVID-19 or other respiratory infections.To determine how COVID-19 pneumonia differs from other types of pneumonia. Information from these studies can potentially help in diagnosis and in furthering our understanding of how SARS-CoV-2 affects the lungs .We present a Convolutional Neural Network in TensorFlow and Keras based Covid-19 pneumonia classification. The proposed system based on CNN using Pneumonia images to classifying the Covid-19, normal, pneumonia this system using CNN model. It is predicted that the success of the obtained results will increase if the CNN method is supported by adding extra feature extraction methods and classify successfully covid-19&pneumonia. We have demonstrated the efficacy and potential of using deep CNN to images

Key Words: COVID-19, Convolutional Neural Network, Kears, TensorFlow and viral pneumonia.

1. INTRODUCTION

The datasetis preprocessed such as Image reshaping, resizing and conversion to an array form. Similar processing is also done on the test image. A dataset consisting of about 19 different plant species is obtained. The train dataset is used to train the model (CNN) so that it can identify the test image and the disease it has CNN has different layers that are Dense, Dropout, Activation, Flatten, Convolution2D, and MaxPooling2D. After the model is trained successfully, the software can identify the disease if the plant species is contained in the dataset. After successful training and preprocessing, comparison of the test image and trained model takes place to predict the disease.

1.1 ARCHITECTURE OF CNN

An input layer, some convolutional layers, some fully-connected layers, and an output layer are elements of a typical architecture of a convolutional neural network. CNN is designed with some modification on LeNet Architecture. Without considering input and output it has 6 layers.

1.1.1 Input Layer

The input layer has fixed dimensions, so the image must be processed before it can be fed into the layer. Normalized gray scale images of size 48 X 48 pixels from given dataset are used for training, validation and testing. For testing propose laptop webcam images are used, in which using OpenCV Haar Cascade Classifier face is detected and normalized.

1.1.2 Convolution and Pooling (ConvPool) Layers

Under batch processing convolutional and pooling is done. Each convolution layer takes image batch input of four dimension N x Color-Channel x width x height. Feature map or filter for convolution is also four dimensional (Number of feature maps in, number of feature maps out, filter width, filter height). In each convolution layer, four dimensional convolutions are calculated between image batch and feature maps. After convolution only parameter that changes is image width and height.

 New image width = Old image width – Filter width+1

 New image height = Old image height – Filter height+1

After each convolution layer down sampling is done for reduction. This process is called Pooling. Max pooling and Average Pooling are two famous pooling methods. After convolution max pooling is done in this project. Pool size of (2x2) is 12 taken, which splits the image into grid of blocks each of size 2x2 and takes maximum of 4 pixels. Two convolution layer and pooling layer are used in the architecture.

1.1.3 Fully Connected Layer

By forward propagation of training data then backward propagation of its errors, the weights of these layers are trained. We can control the training speed and the complexity of the architecture by tuning the hyper-parameters, such as learning rate and network density.

1.1.4 Output Layer

Output from the second hidden layer is connected to output layer having seven distinct classes and output is obtained using the probabilities for each of the seven classes.

2. SYSTEM IMPLEMENTATION

Convolutional neural networks (CNN or ConvNet) is one of the most popular types of deep neural networks. With input data, and uses 2D convolutional layers a CNN convolves learned features, making this architecture well suited to

processing 2D data. Need for manual feature extraction is elimnated by CNN, so you do not need to identify features used to classify images. From images, the CNN works by extracting features directly. The relevant features are not retrained; they are learned while the network trains on a collection of images.

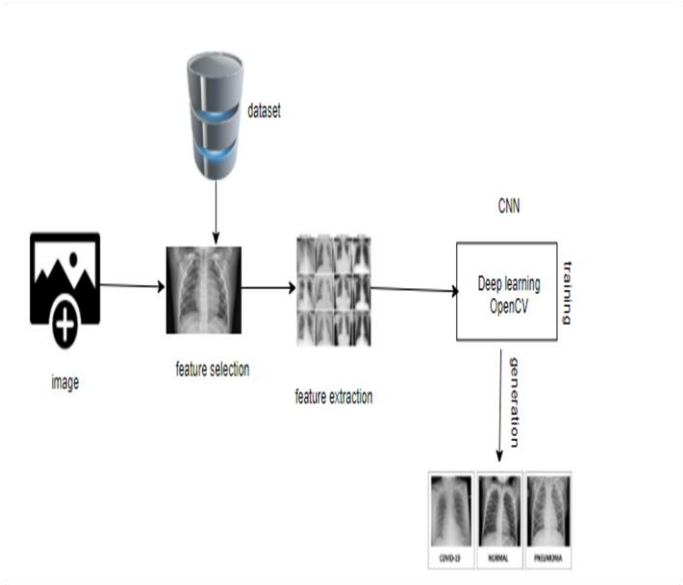

Fig -1: System Architecture

2.2 How to Create and Train Learning Models

2.2.1 Training from Scratch

To train a deep network from scratch, you gather a very large labeled data set that will learn the features and model. This is best for applications, or applications that have a large number of output categories. This is less common approach because, to train it takes a days or weeks.

2.2.2 Transfer Learning

Deep learning applications use the transfer learning approach, that involves fine-tuning a trained model. Start with a existing network, such as AlexNet or GoogLeNet, and feed in new data containing previously an unknown classes. After making tweaks to the network, you can now perform an new task, such as categorizing only dogs or cats instead of 1000 different objects.

2.2.3 Feature Extraction

A slightly less common, more specialized approach to deep learning is to use the network as an feature extractor. Since all the layer are tasked with learning certain features from images, during the training process we can pull these features out of the network at any time. These features can be used as input to an machine learning model such as support vector machines (SVM).

3. MODULES

Detailed description about modules for understanding the system design and working

3.1 MODULE 1: Manual CNN

We have to import our data set using keras preprocessing image data generator function also we create size, rescale, range, zoom range, horizontal flip. Then we import our image dataset from folder through the data generator function. Here we set train, test, and validation also we set target size, batch size and class-mode from this function we have to train using our own created network by adding layers of CNN.

3.2 MODULE 2: AlexNet

To train our dataset using classifier and fit generator function also we can make training steps per epoch's then total number of epochs,validation data and validation steps using this data we can train our dataset.

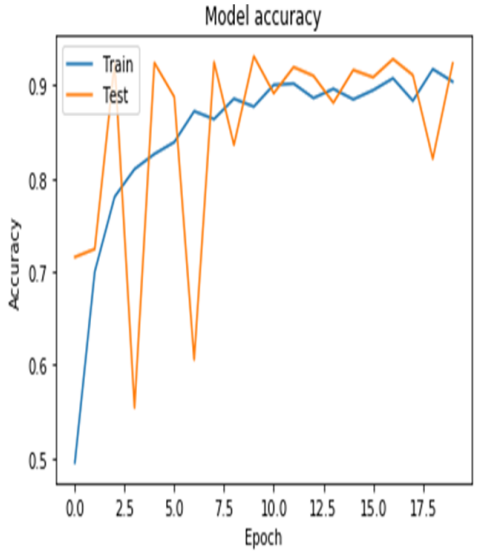

Fig -2: Model accuracy of AlexNet

3.3 MODULE 3: LeNet

Convolutional Neural Network (ConvNet/CNN) is an Deep Learning algorithm which can take in a input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in an ConvNet is much lower compared to other classification algorithms. While in primitive methods filters is hand-engineered, with enough training, ConvNets has the ability to learn these characteristics. The architecture of an ConvNets is analogous to that of the connectivity pattern of Neurons in Human Brain and was inspire by the organization of the Visual Cortex. Individual neurons respond to stimuli only in an restricted region of the visual field known as the Receptive Field.

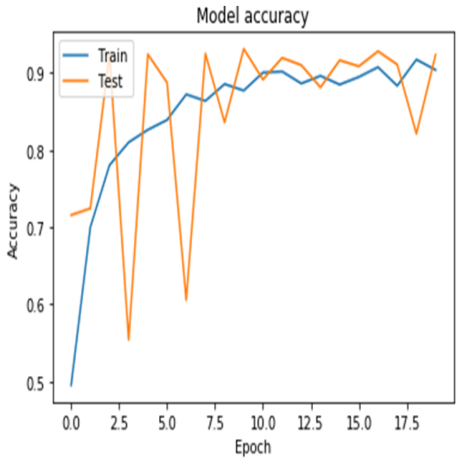

Fig -3 Model accuracy for LeNet

3.4 MODULE 4: Deployment

In this module the trained deep learning models is converted into hierarchical data format file (.h5 file) which then deployed in our django framework for providing better user interface and predicting output whether the given chest X-ray are covid-19 / pneumonia / normal.

Libraries Required

numpy : To process the image matrices

os: To access the file system to read the image from the train and test directory from our machines

random: To shuffle the data to overcome the biasing

matplotlib: To display the result of our predictive outcome.

tensorflow: Just to use the tensor board to compare the loss and adam curve our result data or obtained log.

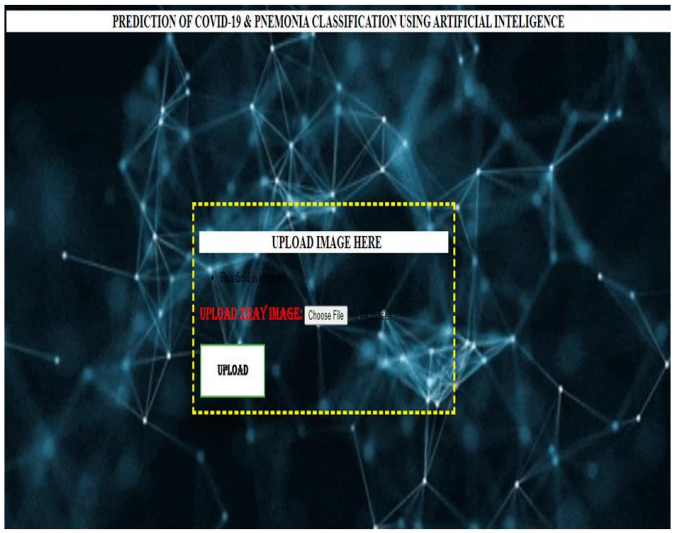

Fig -4 Data uploading page

The above page is the link created using pycharm by which modifying or setting up the path of vies.py in the system coding.

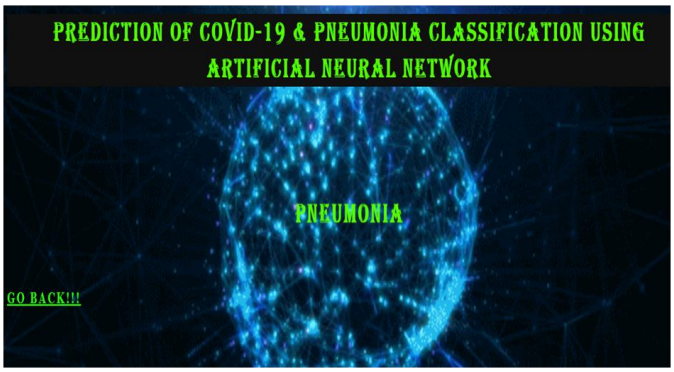

Fig-5 Result Prediction page

4. CONCLUSION

Scientist are working to find medications to treat the disease and to develop an vaccine. The World Health Organization declared the novel corona virus outbreak “a public health emergency of international concern” on January 30. On March 11, 2020 after sustained spread of the disease outside of the China, the World Health Organization declare the COVID-19 epidemic an pandemic. Public health measure like ones implemented in China and now around the world, will hopefully blunt the spread of the virus while treatments and an vaccine is developed to stop it.

4.1 Limitations

- It has not focused on AlexNet CNN in keras and Tensorflow as classifier.
- It cannot train large amount of data at a time.
- It has not focused on increasing the recognition rate and classification accuracy of severity image of chest X-ray.
- It takes more time to predict the results.

4.2 Advantages

- Alexnet CNN is used for classification.
- The large amount ofchest X-ray data can be train on artificial environment.
- Classification accuracy.
- It is best model for deep learning technique to easily classifying Covid-19 & Pneumonia.

4.3 Applications

- In Hospitality, especially even nurse can easily predict and inform the results to patient.

- It can be used anywhere to predict covid-19 without taking test.
- Anyone can access using X-ray to predict in shorter time.

REFERENCES

[1] N. Zhu, D. Zhang, W. Wang, X. Li, B. Yang, J. Song, X. Zhao, B. Huang,W. Shi, R. Lu, P. Niu, F. Zhan, X. Ma, D. Wang, W. Xu, G. Wu, G. F. Gao, and W. Tan, "A novel coronavirus from patients with pneumonia in China, 2019," N. Engl. J. Med., vol. 382, pp. 727–733, 2020.

[2] D. Benvenuto, M. Giovanetti, M. Salemi, M. Prosperi, C. De Flora, L. C. Junior Alcantara, S. Angeletti, and M. Ciccozzi, "The global spread of 2019-nCoV: A molecular evolutionary analysis," Pathog. Glob. Health,pp. 1–4, 2020.

[3] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang, Z. Tang, K. He, Y. Shi, and D. Shen, "Review of Artificial Intelligence Techniques in Imaging Data Acquisition, Segmentation and Diagnosis for COVID-19," IEEE Rev. Biomed. Eng., vol. 3333, no. c, pp. 1–13, 2020.

[4] WHO, "Coronavirus disease 2019 (COVID-19)Situation Report81,"2020.[Online]. Available:https://www.who.int/docs/defaultsource/coronaviruse/situationreports/20200410sitrep81covid19.

[5] F. Hao, X. Luo and C. Mu, "Research on Key Technologies of Intelligent Agriculture Based on Agricultural Big Data," 2016 International Conference on Smart City and Systems Engineering (ICSCSE), Hunan.