



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: IV Month of publication: April 2019

DOI: <https://doi.org/10.22214/ijraset.2019.4541>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Chest X-ray Pneumonia Prediction using Machine Learning Algorithms

S Sreeja¹, L Bhavya², S Swamynath³, R Dhanuja⁴

^{1,3,4}Student, ²Adhoc Lecturer, Dept of CSE, JNTUACEP, Pulivendula, AP, India

Abstract: Predicting whether the person has Pneumonia from his X-ray report of lungs. For this I selected a dataset from Kaggle which was compiled from a wide range of sources and made publicly available by the United States Department of Agriculture Economic Research Service (USDA ERS). So that we can build a model to predict the rate of Pneumonia in a city by collecting the people's X-ray reports. I use a neural network classification model in Keras to classify Normal and Pneumonia persons. Here the input parameters are the training data and the output will either 0 or 1 i.e. Pneumonia or not.

Keywords: Machine Learning, X-ray report of Pneumonia lungs, Convolutional Neural Networks algorithms, accuracy.

I. INTRODUCTION

Pneumonia is an infection in one or both lungs. It can be caused by bacteria, viruses, or fungi. Bacterial pneumonia is the most common type in adults.

Pneumonia causes inflammation in the air sacs in your lungs, which are called alveoli. The alveoli fill with fluid or pus, making it difficult to breathe. According to National Institutes of Health (NIH), chest x-ray is the best test for pneumonia diagnosis. However, reading x-ray images can be tricky and requires domain expertise and experience. It would be nice if we can just ask a computer to read the images and tell us the results.

The main aim of this project is to predict whether the person has Pneumonia from his X-ray report of lungs. For this I selected a dataset from Kaggle. So that we can build a model to predict the rate of Pneumonia in a city by collecting the people's X-ray reports. I use a neural network classification model in Keras to classify Normal and Pneumonia persons. Here the input parameters are the training data and the output will either 0 or 1 i.e. having Pneumonia or not.

II. RELATED WORK

The main aim of my project is to predict whether a person has Pneumonia by examining the X-ray report. For doing this I selected the dataset from Kaggle <https://www.kaggle.com/paultimothymooney/chest-xraypnemunia> Since my goal is to predict the presence of Pneumonia, Here I am using CNN Sequential model in Keras for classification models. My strategy is using this model adding the layers to increase the accuracy. Using convolution layers as needed, Dense layer to avoid overfitting, Flatten layer, MaxPooling layers, series of convolution layers with filters (Kernels), Pooling, fully connected layers (FC) and apply Softmax function.

III. PROPOSED SYSTEM

Using Machine Learning techniques predicting patients pneumonia disease is a time consuming task which degrades patients survival rate, By Applying different machine learning techniques an early diagnosis of lungs problems will increase patients' survival rate based on accuracy, to find the best suitable algorithm for pneumonia prediction which gives best performance. It added a greater advantage to medical field.

The technique that I used is CNN Sequential model in Keras for classification. Technically, deep learning CNN models to train and test, each input image will pass it through a series of convolution layers with filters (Kernels), Pooling, fully connected layers (FC) and apply Softmax function to classify an object with probabilistic values between 0 and 1.

IV. METRICS

The metrics used is AUC. As my dataset is unbalanced, I would like to use auc as Evaluation metric for this categorical classifier. The auc function is defined as

```
def auc(y_true,y_pred):  
    auc=tf.metrics.auc(y_true,y_pred)[1]  
    K.get_session().run(tf.local_variables_initializer())
```

return auc

During development, a validation set was used to evaluate the model. For validation I want to use “categorical_crossentropy” as loss function for CNN,

optimizer as “adam” and also metrics as “auc”.

tf.metrics.auc has auc metrics the arguments used for this are auc(labels,predictions)where labels is Tensor whose shape matches predictions. predictions: A floating point Tensor of arbitrary shape and whose values are in the range [0, 1]

Loss function is an important part in artificial neural networks, which is used to measure the inconsistency between predicted value and actual label.

1) Accuracy = (1 – Error) = (TP + TN)/(PP + NP) = Pr(C), the probability of a correct classification.

2) Sensitivity = TP/(TP + FN) = TP/PP = the ability of the test to detect disease in a population of diseased individuals.

3) Specificity = TN/(TN + FP) = TN / NP = the ability of the test to correctly rule out the disease in a disease-free population.

4) Sensitivity

= TP/(TP + FN) = 160 / (160 + 40) = 80.0% Specificity

= TN/(TN + FP) = 69,860 / (69,860 + 29,940) = 70.0%

V. IMPLEMENTATION

A. Data Loading

Data is loaded using CV2 package in python. For loading the images from the dataset. The dataset consists of three folders testset, train set and validation set.

Total 1583 Normal images and 4273 Pneumonia images are present. There are 5856 total images ,624 test set images,4236 test set images,16 validation set images.

B. Model Building

Sequential model with 4 convolution layers,2 max pooling layers,2 batch normalization layers, one dense layer, one flatten layer, one drop out layer is built.

C. Data Preprocessing

As in implementation model I have created this model using CNN Sequential model.

D. Training

In this model four convolution layers are used with filters=32,32,64,64 and activation as “relu”, padding as “same”, and two maxpooling layers are used with pool size=(2,2),one flatten layer, two batch normalization layers, two dropouts with 0.25 order to avoid overfitting, Dense layer with 2 units and activation function as “softmax”.

E. Dataset Description

Classifying the X-ray images as Pneumonia and Normal Dataset contains totally 5856 images. This data set contains 5856 X-rays in which 1583 are normal images and 4273 are Pneumonia images. Photos are not high resolution. Photos are not reduced to a single size, they have different proportions. The total dataset is divided into validation set, testing set, training set. ((8,8),(234,390),(1341,3875) respectively).

F. Data Preprocessing

First task is to read the dataset and perform visualizations on it to get some insights about the data. I want use the cv2 library for the reading the dataset and resize the (64*64*1) since it is RGB image. The corrupted images are not read they are skipped. After Data Exploration, as already the data is split into training, validation and testing sets and normalized, the data now is suitable for Convolutional neural networks. Since I have large data set so don't need for data augmentation in case if I used the data augmentation there may be some memory issues so I want to avoid it.

VI. CLASSIFICATION TECHNIQUES

A. CNN (Convolution Neural Network)

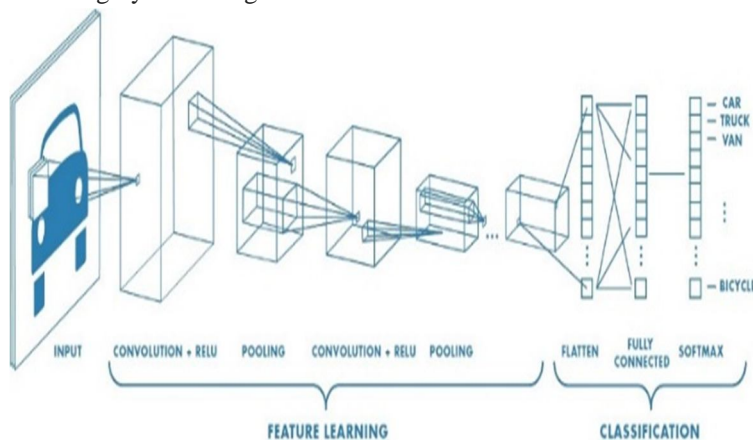
A convolutional neural network (CNN) is a specific type of artificial neural network that uses perceptrons, a machine learning unit algorithm, for supervised learning, to analyze data. CNNs apply to image processing, natural language processing and other kinds of cognitive tasks. A convolutional neural network is also known as a ConvNet. Like other kinds of artificial neural networks, a convolutional neural network has an input layer, an output layer and various hidden layers. Some of these layers are convolutional, using a mathematical model to pass on results to successive layers. This simulates some of the actions in the human visual cortex. CNNs are a fundamental example of deep learning, where a more sophisticated model pushes the evolution of artificial intelligence by offering systems that simulate different types of biological human brain activity. A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, RELU layer i.e. activation function, pooling layers, fully connected layers and normalization layers.

The technique that I used is CNN Sequential model in Keras for classification. CNNs have wide applications in image and video recognition, recommender systems and natural language processing.

In neural networks, Convolutional neural network (ConvNets or CNNs) is one of the main categories to do images recognition, images classifications. Objects detections, recognition faces etc., are some of the areas where CNNs are widely used.

CNNs, like neural networks, are made up of neurons with learnable weights and biases. Each neuron receives several inputs, takes a weighted sum over them, pass it through an activation function and responds with an output. The whole network has a loss function and all the tips and tricks that we developed for neural networks still apply on CNNs.

CNN image classifications takes an input image, process it and classify it under certain categories (Eg: Dog, Cat, Tiger, Lion). Computers sees an input image as array of pixels and it depends on the image resolution. Based on the image resolution, it will see $h \times w \times d$ (h = Height, w = Width, d = Dimension). Eg: An image of $6 \times 6 \times 3$ array of matrix of RGB (3 refers to RGB values) and an image of $4 \times 4 \times 1$ array of matrix of grayscale image.



B. Convolution Layer

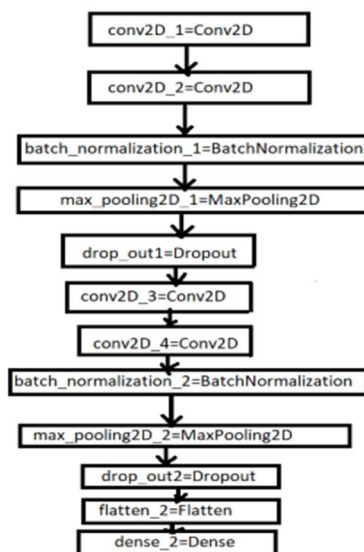
Convolution is the first layer to extract features from an input image. Convolution preserves the relationship between pixels by learning image features using small squares of input data. It is a mathematical operation that takes two inputs such as image matrix and a filter or kernel.

C. Pooling Layer

Pooling layers section would reduce the number of parameters when the images are too large. Spatial pooling also called subsampling or downsampling which reduces the dimensionality of each map but retains the important information. MaxPooling Layer is used in this project.

D. Fully Connected Layer

The layer we call as FC layer, we flattened our matrix into vector and feed it into a fully connected layer like neural network.



VII. RESULTS

Since the size of dataset is small at present , there is not much difference between training and testing times of different algorithms.

A. ROC Curve

```

tpr,fpr,thres = roc_curve(pre["label"],pre[1])
roc = pd.DataFrame([tpr,fpr]).T
roc.plot(x=0,y=1)
  
```

B. Confusion Matrix

```

from import classification_report,confusion _matrix
from mlxtend.plotting import plot_confusion_matrix
y_pred = np.argmax(p, axis=1)
CM = confusion_matrix(test_batches.classes,y_pred)
fig, ax = plot_confusion_matrix(conf_mat=CM , figsize=(5, 5))
plt.show()
  
```

VIII. CONCLUSION

In this project, we have proposed methods for predicting pneumonia in patients using machine learning techniques. The machine learning techniques that were used include deep learning which we can say Convolutional Neural Networks. The system was implemented using all the models and their performance was evaluated. Performance evaluation was based on certain performance metric. The trained model was compiled with “adam” as optimizer, loss function is “categorical_crossentropy”, metrics as “auc”. The final accuracy is around 97% after all 4 final epochs.

REFERENCES

- [1] <https://keras.io/models/sequential/>
- [2] <https://medium.com/technologymadeeasy/the-best-explanation-ofconvolutional-neural-networks-on-the-internet-fbb8b1ad5df8>
- [3] <https://machinelearningmastery.com/keras-functional-api-deeplearning/>
- [4] Datasetlink <https://www.kaggle.com/paultimothymooney/chest-xraypneumonia>.
- [5] <https://www.meded.virginia.edu/courses/rad/cxr/pathology3chest.html>
- [6] <https://www.drbeen.com/blog/the-basics-of-chest-x-rayinterpretation/>
- [7] <https://keras.io/getting-started/sequential-model-guide/>
- [8] <https://machinelearningmastery.com/keras-functional-api-deep-learning/>
- [9] https://brohrer.github.io/how_convolutional_neural_networks_work.htm



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)