Proceedings of Pneumonia Prediction with Tensorflow and CNN
December 2021, , ,

PNEUMONIA DISEASE PREDICTION USING TENSORFLOW

Alhamdou Jallow
Antern CS001 Student
Computer Science And Engineering Student
Email: jallowalhamdou.9@gmail.com

ABSTRACT

In this project we explore how to used Deep Learning with the help of tensor flow in order to predict if a patient is having Pneumonia or not with the help of existing data. Using Deep Learning Algorithms and some libraries Tensorflow. Based on the Libraries the VGG16 was able to give a better prediction that the plain CNN

INTRODUCTION

What is Pneumonia? Pneumonia is a form of acute respiratory infection that affects the lungs. The lungs are made up of small sacs called alveoli, which fill with air when a healthy person breathes. When an individual has pneumonia, the alveoli are filled with pus and fluid, which makes breathing painful and limits oxygen intake

How do Deep Learning Networks distinguish between healthy lungs

Most deep neural network applied to the task of pneumonia diagnosis have been adapted from natural image classification. These models have a large number of parameters as well as high hardware requirements, which makes them prone to over fitting and harder to deploy in mobile settings. Some research on medical image classification by CNN has achieved performances rivalling human experts. This Model helps in predicting if an individual is having pneumonia or not. The model has been train in numerous data collected from kaggle repositories which was donated by reputable doctors. Deep Learning helps us in various domains of knowledge ranging from health care, transportation, real estates. The models are always train on clear and clean data collected either from clients or from through scrapping sites.

1 Dataset and Features

The dataset was taken from kaggle from this link https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia The dataset comprises of two train, test, and validation files, each file is sub divided into two other sub folder each folder containing train and test set of images. The train folder is used to train the dataset while the test folder is used to test the validation folder is used in validating the model that was build. However, some other image preprocessing was done with the help of Tensorflow to make sure that the model will be able to do the prediction in clean and clear data just for the model to be able to do its prediction better. .

2 Methods and Data Processing

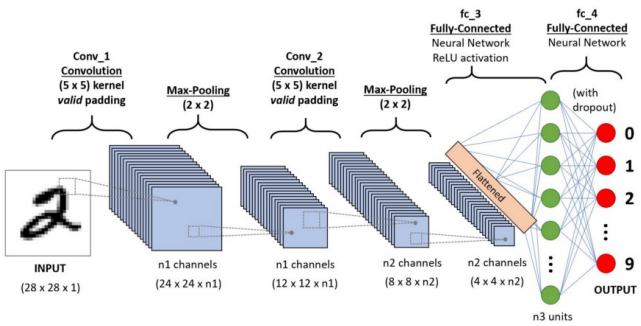
Deep Learning never understands numbers alone. The Data has to be process into something valuable to it for the machine to be able to do its prediction accurately through machine codes. preprocessing.**imagedatasetfromdirectory**, this is a method that was called In-order to help in the processing of data. It helps in setting the image size to the chosen size of the developers it also helps in setting the **colorcode** of the image either in **"rgba"**,"rgb", **"binary"**. it also helps in identifying the number of classes present in all. In this method the dataset is divided in to train, test and validation so it works in all this directories. Image Data Generator is also a method that is used on processing data, it helps in re-scaling and resizing of our data. The images are resize into 1/255.0 in-order to used A preprocessing layer which rescales input values to a new range.To rescale an input in the [0, 255] range to be in the [0, 1] range, you would pass scale=1./255. The rescaling is applied both during training and inference. Data Augmentation is also done to make sure that after the model is

put into production any image that wants to be predicted it can be any form, RandomFlip(Horizontal and vertical), Random Rotation in any kind of angle.

3 Models

3.1 Convolutional Neural Network

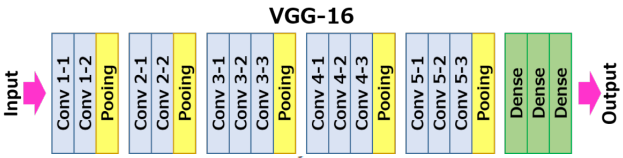
convolutional neural network (CNN/ConvNet) is a class in deep neural networks, commonly applied in analyzing visual images.Deep Learning helps in recognizing objects in images by using a CNN. CNNs are playing a major role in diverse fields e.g image processing problems, computer vision tasks like localization and segmentation, video analysis, to recognize obstacles in self-driving cars. As CNNs are playing a significant role in these fast-growing and emerging areas, they are very popular in Deep Learning. Convolutional Neural Network has fundamentally changed our approach towards image recognition as they can easily detect patterns and understand them. Using the CNN and some Evaluation metrices, Accuracy, Precision, Recall and AUC. With this evaluation metrices was able to have a prediction of accuracy score of 94.52, Precision score of 96.21, loss score of 0.15.



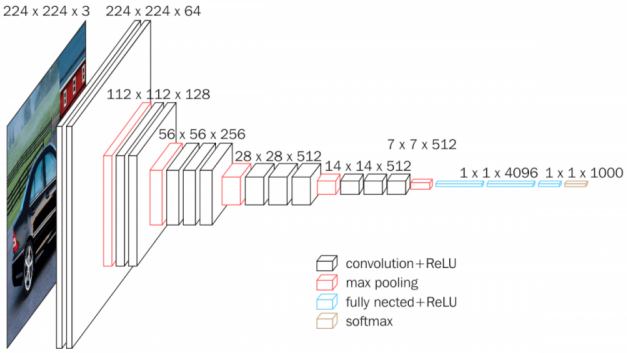
4 VGG16 Architecture

VGG16 is a convolution neural net (CNN) architecture which was used to win ILSVR(Image-net) competition in 2014. The input to the network is image of dimensions (224, 224, 3). The first two layers have 64 channels of rgb 3*3 filter size and the same padding. After a max pool layer of strides (2, 2), two layers which have convolution layers of 256 filter size and filter size (3, 3). This followed by a max pool layer of stride (2, 2) which is same as previous layer. There are 2 convolution layers of filter size (3, 3) and 256 filter. After that there are 2 sets of 3 convolution layer and a max pool layer. Each have 512 filters of (3, 3) size with same padding.This image is then passed to the stack of two convolution layers

4.1 Configuration of VGG



ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					



5 Splitting The Data

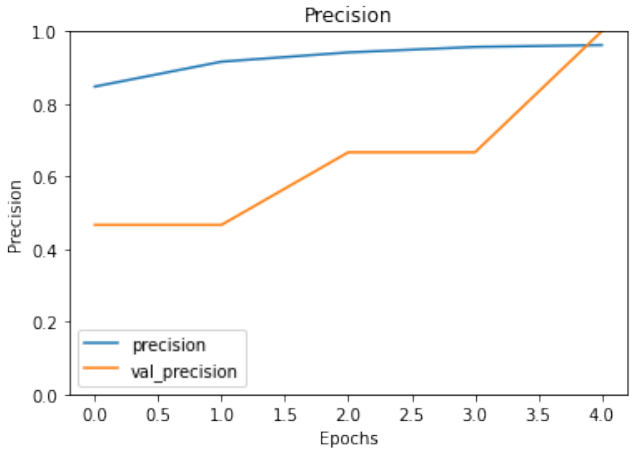
The dataset was split into a train and test and validation data. However, the dataset came with custom folders of train of test folders and validation. The train folder was used to build and train the model so that it can recognized patterns while the test dataset was used to test the model that was built. However, in order to avoid over-fitting we divided the dataset into train and test dataset. furthermore we model is evaluated with the help of Precision, Recall and Accuracy on the test dataset. The train and test model is visualize with the help of graph.

6 Results

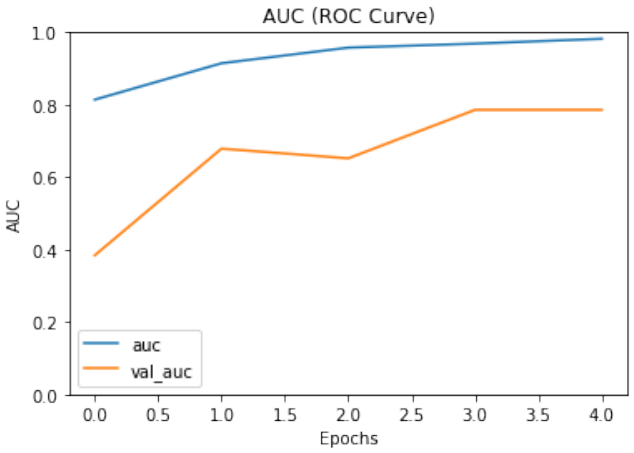
The results were quiet amusing looking at the model prediction we used to test our CNN and VGG Architecture. The Model

with least performance was the plane CNN model with an accuracy score of 81 with, precision score of 84. It was rather suprising that the VGG16 model has a higher accuracy score compare to the normal CNN model .

6.1 Precision

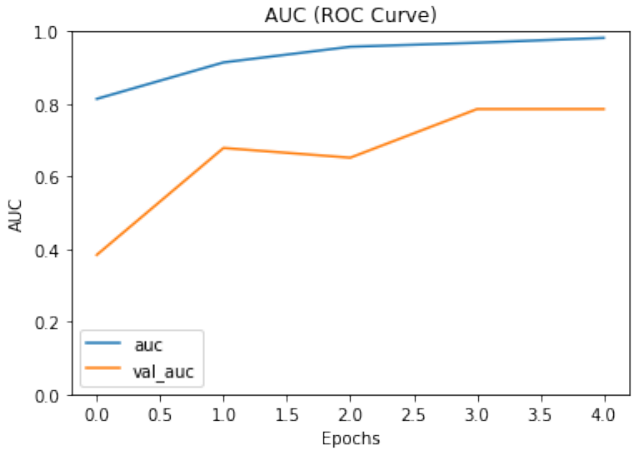


6.2 Accuracy



6.3 Area Under the Curve

6.4 Precision



7 Conclusion

This project has widen my knowledge on how to implement various Deep learning Algorithm specifically the CNN model and how to optimize them in real life. it has also exposed me into the world of Deep Learning which is a quiet amazing journey. Looking at the various Algorithms used in this project it can be seen that CNN Architecture VGG16 has outperform all the other Architectures. Notwithstanding in the near future i hope to used more sophisticated Algorithms and Architectures that can help me to have more better accuracy than the one i have in this version of my project

8 Acknowledgement

I Alhamdou will take this opportunity to thank Ayush for his tremendous efforts in helping me elevate myself in the field of machine learning and Deep Learning he has been of great support to me in the entire duration of the course and a special thanks to all my fellow batch students of the Antern CS001.

9 Reference

https://arxiv.org/pdf/1409.1556.pdf https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cn/ http://www.linliang.net/wp-content/uploads/2018/01/AAAI2018$_{W}AE.pdf$https : //neurohive.io/en/

https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c

https://medium.com/analytics-vidhya/vggnet-architecture-explained-e5c7318aa5b6 https://keras.io/api/applications/usage-examples-for-image-classification-models