# Title:

Solution Design Mission

Navil Hassan (W19015723)

Team: Count Dooku

Table of Contents

[Title: 1](#_Toc71523052)

[Introduction 3](#_Toc71523053)

[Solution Diagram 3](#_Toc71523054)

[Solution Description 4](#_Toc71523055)

[Data Description 4](#_Toc71523056)

[Solution Motivation 5](#_Toc71523057)

[References 5](#_Toc71523058)

# *Introduction*

Artificial Intelligence has the potential to revolutionise disease diagnosis by implementing an automated detection system using a deep learning algorithm. This project is experimentation to create a model that uses static images of chest x-rays to detect pneumonia on an early stage using Artificial Intelligence. To achieve this we have used this proposed solution to create the API (Application Programming Interface) which delivers a user an accurate response to a system and sends the request back to the user and a simple server to gather all data necessary for the pneumonia and normal datasets.

# *Solution Diagram*

1. Start API: input image -> Pre-process Image -> Data augmentation -> train model -> make prediction.

Start API

Input image

Pre-process image

Make Prediction

Data augmentation

Train model

**Figure 1**

1. Server: Server <--- Commands --> Client.

Server

Commands

**Figure 2**

Client

# *Solution Description*

I have designed a couple of models which is capable of classifying x-rays images of lungs that detects pneumonia or not within the lungs of a patient. To achieve this, two solution models were proposed to make the API fully functional with alongside a server as it is shown above **(Figure 1 & 2).**

**Figure 1** illustrates if the patient is diagnosed with pneumonia or not by processing images and data augmentation. Then, the model is trained to make an prediction to give an accurate result of the patient. This solution is written in Python.

**Figure 2** illustrates a server which was also written in Python which accepts several commands to be able to receive images and make a prediction of the image then return the correct accurate result. This executes the prediction on the server itself with the suitable hardware and allow users to run the API and use the model on any hardware e.g. Tablets, Smartphones, desktops and Laptops.

Both solution combined created a simple working system which detects pneumonia by classifying x-rays images.

# *Data Description*

Datasets were gathered from Kaggle, we have chosen range of seven models to go with in order to give us the best possible accuracy to test against for predictions. The model attains 93% accuracy when all the models are combined while C-NN performed at 84% (Figure 3) accuracy. Ensembling became possible as the method of connecting the performances of all the models to provide a higher accuracy of 93% against the test set and 96% with the optional validation set.

Accuracy of each model used on small tests:

|  |  |
| --- | --- |
| **Models:** | **Accuracy %** |
| Xception | 92% |
| Vgg19 | 90% |
| Resnet | 92% |
| Inception | 92% |
| Densenet | 93% |
| C-NN | 84% |
| Esembled | 93% |

**Figure 3**

Accuracy of each model used on greater tests:

|  |  |
| --- | --- |
| **Models:** | **Accuracy %** |
| Xception | 97% |
| Vgg19 | 95% |
| Resnet | 94% |
| Inception | 97% |
| Densenet | 95% |
| C-NN | 93% |
| Esembled | 96% |

**Figure 4**

Conversly, When the C-NN model was tested against a greater test set of 2268 images. The accuracy of our model (C-NN) enhanced to 93% (Figure 4) which is a magnificent jump from the accuracy it got from Figure 3. This shows how reliable it is when compared to greater dataset. Also, on inspection model of the Kaggle dataset, it was noted to be small which made creating an accurate model problematic, and with impractical way to inspect the dataset during collection, it was decided to create new samples using data augmentation to create a greater dataset for the training.

# *Solution Motivation*

The motivation behind the final proposed solution was to make an reliable and simple API which our target audience an young/middle/old age adult (20-60 years old) with minimal IT knowledge can use on their devices to if they have pneumonia or not by using the x-ray images. Also, this API can be used in Hospitals to detects its patient as this is less resourceful to maintain and highly beneficial to medical institutions.

# References

Jang, A., 2021. *TensorFlow Pneumonia Classification on X-rays*. [online] Kaggle.com. Available at: <https://www.kaggle.com/amyjang/tensorflow-pneumonia-classification-on-x-rays/data#3.-Visualize-the-dataset> [Accessed 17 April 2021].

Hassan, M., 2021. *VGG16 - Convolutional Network for Classification and Detection*. [online] Neurohive.io. Available at: <https://neurohive.io/en/popular-networks/vgg16/> [Accessed 8 April 2021].

Zhang, X., Zou, J., He, K. and Sun, J., 2021. *Accelerating Very Deep Convolutional Networks for Classification and Detection*. [online] arXiv.org. Available at: <https://arxiv.org/abs/1505.06798> [Accessed 8 April 2021].

Thabo, M., 2021. *99+% Accurate CNN that detects Pneumonia*. [online] Kaggle.com. Available at: <https://www.kaggle.com/joythabo33/99-accurate-cnn-that-detects-pneumonia> [Accessed 9 April 2021].

Tripathi, A., 2021. *Xception Model for Pneumonia Detection🤧*. [online] Kaggle.com. Available at: <https://www.kaggle.com/achintyatripathi/xception-model-for-pneumonia-detection#Model-Creation> [Accessed 10 April 2021].