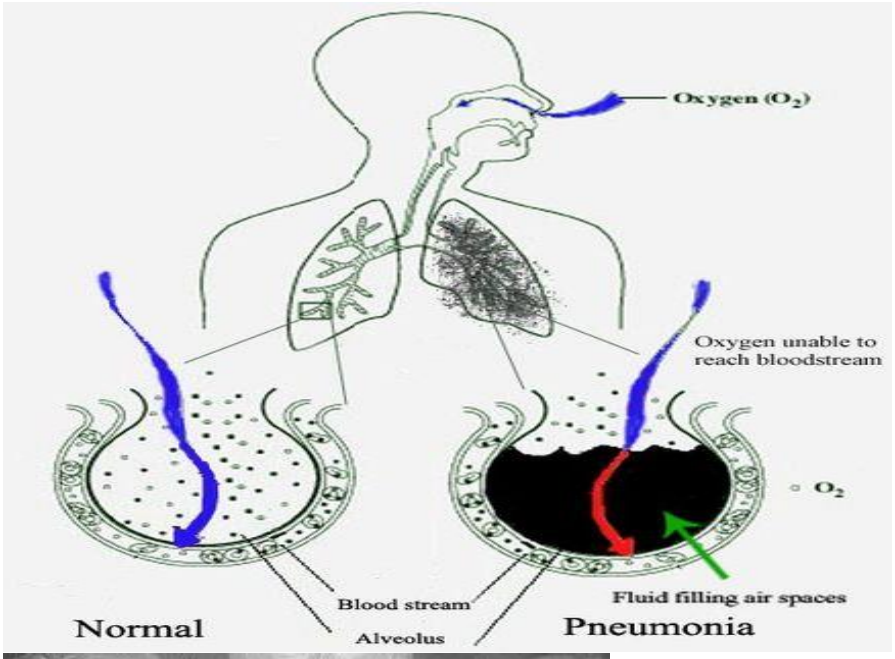
Capstone Project Proposal

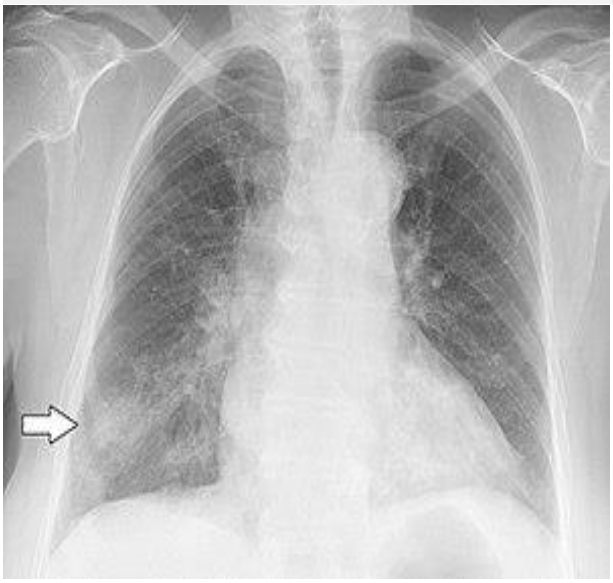
Domain Background: Pneumonia

is an inflammatory condition of the lung affecting primarily the small air sacs known as alveoli, Typically symptoms include some combination of productive or dry cough, chest pain, fever, and trouble breathing. Severity is variable.

Pneumonia is usually caused by infection with viruses or bacteria and less commonly by other microorganisms, certain medications and conditions such as autoimmune diseases. Risk factors include other lung diseases such as cystic fibrosis, COPD, and asthma, diabetes, heart failure, a history of smoking, a poor ability to cough such as following a stroke, or a weak immune system. Diagnosis is often based on the symptoms and physical examination. Chest X-ray, blood tests, and culture of the sputum may help confirm the diagnosis. The disease may be classified by where it was acquired with community, hospital, or health care associated pneumonia.

There is a detailed information in https://en.wikipedia.org/wiki/Pneumonia.





Pneumonia affects approximately 450 million people globally (7% of the population) and results in about 4 million deaths per year, we think that it's a must to use our knowledge to help those people. Of course there is a lot of research on this, and there is some Neural Networks ready to use but we tried to work on a new way so maybe we can build a better model.

The more detailed information for pneumonia can be found in the following research paper link https://www.papermasters.com/pneumonia.html.

Application: Diagnosis of Pneumonia by visualizing the X-ray reports in hospitals. Also helps to find out the rate of Pneumonia disease in a city by collecting the X-ray reports of people.

Problem Statement: 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 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. having Pneumonia or not.

Datasets and Inputs: The dataset is downloaded from https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia.

This dataset contains 3 sub folders with necessary information to make predictions. They are test,train,val. All together contains 5856 images.

test folder contains total 624 images classified into 2 classes namely Normal and pneumonia contains 234,390 images respectively.

val folder contains total 16 images classified into 2 classes namely Normal and pneumonia contains 8,8 images respectively.

train folder contains total 5216 images classified into 2 classes namely Normal and pneumonia contains 1341,3875 images respectively. All the images do not have fixed size and high resolution.

Solution Statement: The classifier is a Convolutional Neural Network, which contain SoftMax activation function and I want to use adam optimizer. I explore the data set with opencv and matplotlib.pyplot libraries. In this project the following parameters can be tuned to optimize the classifier:

Hyper parameters -

1. learning rate alpha

2. Number of layers

3. Mini-batch size

Benchmark Model: In this classification, I want to use random assignment to set the worst score benchmark. With one convolution layer and one maxpooling layer and one dense with 2 unit that classifies the data with minimum 50% accuracy. Now I will try to build a model which has an accuracy more than the Benchmark model.

Evaluation Metrics:

I want to use auc as evaluation metric for Pneumonia classification. As my dataset is imbalanced dataset it will be best metric for categorical classifiers.

I defined auc function

```
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
```

and used it in Model.compile(Adam(lr=0.001),loss="categorical_crossentropy", metrics=[auc])

During development, a val folder which is containing 8 images is to be used as validation set. For validation I want to use “categorical_crossentropy” as loss metric for CNN, optimizer as “adam” and also metrics as “auc”.

Project Design:

The project is composed of different steps as follows :-

Pre-Processing:-

- 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.

Training:

- I want to apply Convolutional Neural Networks of my own and use on the data.
- I want to apply the neural network sequential model in Keras developed for a two class (binary) classification model.
- I would like to apply Sequential CNN model in keras.
- We train a deep convolutional neural network on a dataset of images, during the training process, the images are passed through the network by applying several filters on the images at each layer. The values of the filter matrices are multiplied with the activations of the image at each layer. The activations coming out of the final layer are used to find out which class the image belongs to.
- I want to use 2D convolution layer as convolution layer for my keras model.
- I first try to add layer one by one on my basic knowledge and after observing the accuracy of the model. I try adding new layers until I reaches a maximum accuracy such that I can't increase my accuracy anymore.
- Also want to develop a model which uses maxpooling2D as pooling layer also whichuses core layer such as Dropout, Dense, Flatten layer and BatchNormalization as normalization layer which Normalize the activations of the previous layer at each batch, i.e. applies a transformation that maintains the mean activation close to 0 and the activation standard deviation close to 1.

- I try to add each layer and after observing the accuracy of the model.I try adding new layers such that the accuracy should be greater than the previous.

Testing:

- The model is tested against the test images folder. I want to train and test the model in Kaggle which provides both CPU and GPU for training , RAM and DISK for Storing the data.