











dragunat2016 /  
Pneumonia\_Image\_Detection




[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)



Repository for Detecting Pneumonia from X-Rays off lungs


 0 stars  0 forks  1 watching  Activity

 Public repository



 main ▾

...

 Branches  Tags

 dragunat2016 Modified README.ME format ... 29 minutes ago ⌚ 8

[View code](#)

 README.md 

# Pneumonia\_Image\_Detection

## Background

Pneumonia is an infection that inflames the lungs from many potential vectors: Bacteria, Viruses, Fungi, etc. Pneumonia is a potentially fatal infection, that if not identified early on could put the patient's life at risk especially in acute-care settings.

Data sets off lungs with Pneumonia and healthy lungs were provided by Guangzhou Women and Children's Medical Center, Guangzhou.


 [image.png](#)

Figure S6. Illustrative Examples of Chest X-Rays in Patients with Pneumonia, Related to Figure 6 The normal chest X-ray (left panel) depicts clear lungs without any areas of abnormal opacification in the image. Bacterial pneumonia (middle) typically exhibits a focal lobar consolidation, in this case in the right upper lobe (white arrows), whereas viral pneumonia (right) manifests with a more diffuse "interstitial" pattern in both lungs.

## Repository Structure

- README
- index.ipynb: Main notebook containing analysis and calculations

- Models: Contains CNN models generated during study
- presentation.pdf: Presentation on this subject
- chest\_xrays: Contains Dataset for project

## Data Structure [↗](#)

---

The data is stored in chest\_xrays. The folder has 3 sub-folders called train, test, and val.

The data is distributed in a very uneven manner across the three.

- Number of training samples: 5216
- Number of testing samples: 624
- Number of validation samples: 16

As a result, all records were combined then split into training and testing data sets. Process described below under data preparation.

Each off these folders had two folders. They were called 'NORMAL' and 'PNEUMONIA'.

These folders had the images, stored in JPEG off the lungs.

- chest\_xray
- test
- NORMAL
- Images of normal lungs
- PNEUMONIA
- Images of lungs with pneumonia
- train
- NORMAL
- Images of normal lungs
- PNEUMONIA
- Images of lungs with pneumonia
- val
- NORMAL
- Images of normal lungs
- PNEUMONIA
- Images of lungs with pneumonia

## Objectives [↗](#)

---

The goal is to create a model that accurately predicts if an X-ray of a lung has pneumonia. The model will optimize on accuracy because we can save lives by accurately predicting lungs that have pneumonia.

The model will be deployed in a hospital where time is critical. Identifying someone who has pneumonia hours before complications occur can save lives.

This might mean that false positives are identified, scenarios where the model predicted that an x-ray has pneumonia when it really did not. This is ok in the business context because hospital staff can view other symptoms or other lab data to confirm if pneumonia is present or not.

## Data Source and Exploration

---

The total data size has over 5000 images across a training, testing, and validation set. The total size is around 2 gigabits. Due to the size, the data was not pushed to the repository.

To access data download the set from this link: <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia>

Data contains 3 data sets, train, test, and validation. In each file, there are two folders: Pneumonia and normal. Images in the Pneumonia file are named depending on whether the source of the infection is bacterial or viral.

First separate them out based on whether they are normal or have pneumonia.

The images that contain pneumonia are divided based on whether the cause of the ailment was bacterial or viral. Since the purpose of the model is to determine whether Pneumonia is present, we will not distinguish or aim to predict the source of the pneumonia.

However, this distinction will come into play later in the analysis when we investigate the parts of the image the model used to determine if pneumonia is present or not. There is a difference in what part of the lung the model uses to distinguish between the two depending on the source.

## Data Preparation

---

The Process to prepare the data was as follows:

1. First we used Image Data Generator on the images in the test, train, and val data.
2. Then we created the labels for the target variable. Whether an xray is for a patient with Pneumonia or not.
3. Due to the disparities between the given training, testing, and val set, all data was combined into one dataset.
4. Data then split into training and testing sets. Since Neural Networks are computationally expensive. Two train test splits were done to reduce the size. The final training data set contained nearly 2500 images.

## Modeling

---

Four models were evaluated through an iterative approach.

### Convolution Neural Network 1

---

Base model is a simple convolution neural network. Contains one convolution layer, one pooling layer, and 10 neurons in the dense layer.

This model has achieved a ~96% accuracy on the training data set and a 95% accuracy on the testing data set.

## Convolution Neural Network 2 [↗](#)

---

This network contained:

- Three Convolutional Layers and three pooling layers
- Dense Layer contains 30 neurons

Accuracy of training and testing data is 96%.

## Convolution Neural Network 3 [↗](#)

---

- Three Convolution layers
- Three Pooling layers
- 30 neurons in the dense layer
- Added a dropout parameter of 0.25

Adding a dropout rate did not significantly change the accuracy of the models. The training accuracy was 94% and the testing accuracy was 95%.

## Convolution Neural Network 4 [↗](#)

---

- Three Convolution layers
- Three Pooling layers
- 30 neurons in the dense layer
- Added a dropout parameter of 0.25
- L2 regularizer parameter of 0.001 Added

Training Accuracy is 93% and testing accuracy is 95%. The training accuracy dropped somewhat.

## Final Model Evaluation [↗](#)

---

Based on the accuracies above, the first neural network with one convolution layers and no regularizers or drop out had the highest accuracy.

Ultimately, all models had accuracies within a percentage point from one - another. It would not be surprising if one of the models with regularization performed better on a different data set with significantly more records.

If this were based on accuracy alone then the first convolution neural network would be the choice.

However, the first, second, and third models had lower recalls. The fourth models had a higher recall by nearly 3-6 percentage points.

In addition, when reviewing the plots of the accuracy/loss over the epochs, we noticed that the models without any regulation had significant variance than the ones that did not.

Given the fact that the fourth model had a higher recall and less variability, the fourth model is the one we choose.

 [Accuracy\\_Epoch](#)

 [Accuracy\\_Models](#)

## Feature Analysis [↗](#)

---

 [Bacteria](#)

 [Viral](#)

The above images display the feature maps generated by each convolutional layer. These maps identify unique patterns.

## Conclusion [↗](#)

---

- Further investigate using the model too evaluate pneumonia in chest - xrays. 95% accuracy is very high and this metric means that \* further investigation is warranted.
- Regularization helped in reducing the variability off the accuracy in later epochs, but reduced the accuracy. These models need to be tested on larger data sets to further investigate the effects off regularization.
- Start running models on a small scale at hospitals.

## Contact Information [↗](#)

---

- Github Link: [https://github.com/dragunat2016/Pneumonia\\_Image\\_Detection](https://github.com/dragunat2016/Pneumonia_Image_Detection)
- Linkedin: <https://www.linkedin.com/in/dhruv-ragunathan-908993b1/>



---

### Releases

No releases published

[Create a new release](#)

---

### Packages

No packages published

[Publish your first package](#)

---

### Languages

● Jupyter Notebook 100.0%