# Detecting Pneumonia on Chest X-Ray using Convolutional Neural Network

Final Capstone Project

Index



## Problem Statement

The goal of this project is to develop a convolutional neural network (CNN) capable of classifying chest X-ray images into two categories: normal (no pneumonia) and pneumonia (infected). This model aims to assist healthcare professionals in diagnosing pneumonia more efficiently and accurately, ultimately improving patient outcomes by facilitating timely and appropriate treatment.

## Context

Pneumonia is a serious respiratory infection that can be life-threatening, especially in vulnerable populations such as the very young, elderly, or those with weakened immune systems. Early and accurate diagnosis is critical for effective treatment. Traditional diagnostic methods, such as chest X-ray analysis, require significant expertise and time, which can delay treatment.
Leveraging machine learning, particularly CNNs, to automate the classification of chest X-ray images presents an opportunity to enhance diagnostic accuracy and speed, potentially saving lives and optimizing healthcare resources.

# Criteria of Success

- Accuracy: The model should achieve high accuracy in classifying chest X-ray images, ideally exceeding 90%.
- Precision and Recall: High precision and recall in detecting pneumonia cases to minimize false positives and false negatives.
- Robustness: The model should generalize well to new, unseen data, demonstrating robustness across different patient demographics and image qualities.
- Efficiency: The model should be computationally efficient, enabling rapid analysis of chest X-ray images in a clinical setting.
- Ease of Use: The end-user interface should be intuitive and easy to use for clinicians with varying levels of technical expertise.

# Constraints

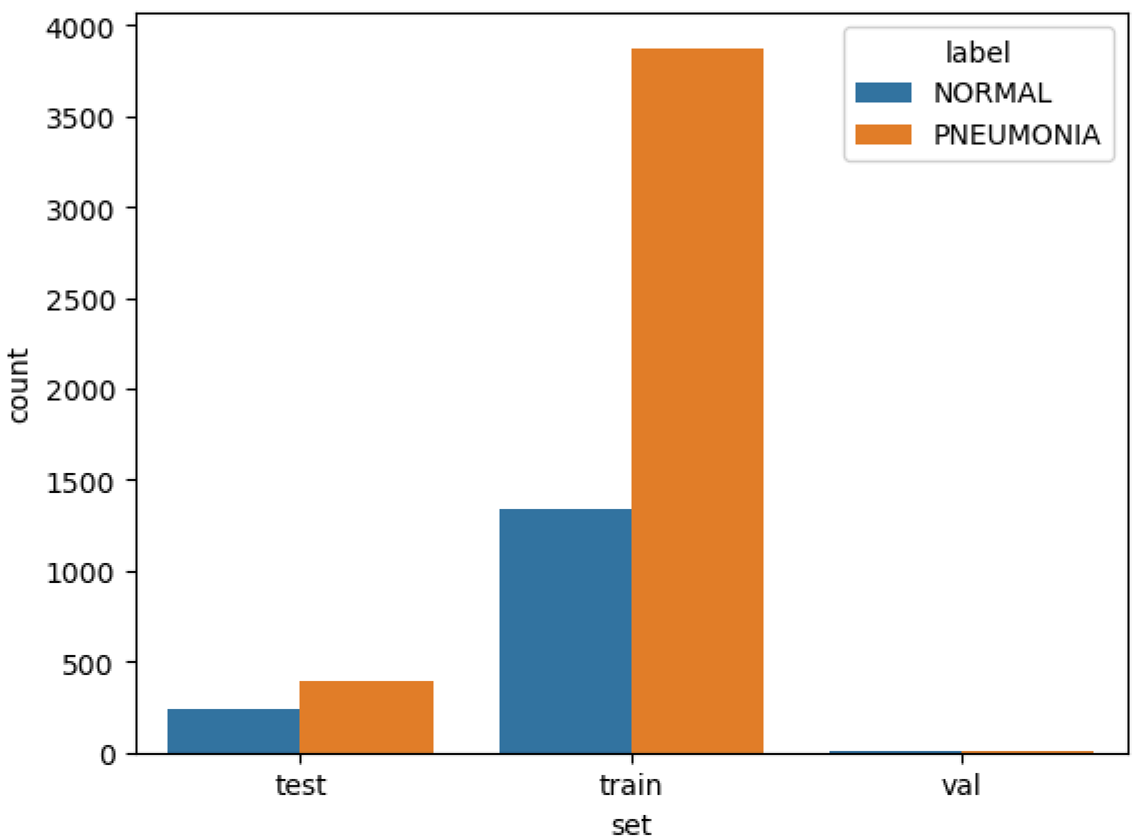
- Data Imbalance: The initial dataset has an imbalance between normal (1341 images) and pneumonia (3875 images) cases, which could bias the model. Addressing this imbalance through techniques like oversampling, undersampling, class weighting or finding more images will be necessary.
- Computational Resources: Training CNNs is resource-intensive, requiring access to powerful GPUs and sufficient memory.
- Data Quality: Variations in image quality and the presence of artifacts can affect model performance. Effective preprocessing and augmentation strategies are essential.
- Ethical Considerations: Ensuring that the model does not inadvertently learn biases related to demographic or socio-economic factors.

# Data

## Data Sources

The initial dataset (https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia) has a total of 5863 images organized directory-wise as:

```
chest_xray/
├───test/
│   ├───PNEUMONIA/ -> 234 images
│   └───NORMAL/    -> 390 images
├───train/
│   ├───PNEUMONIA/ -> 3875 images
│   └───NORMAL/    -> 1341 images
└───val/
    ├───PNEUMONIA/ -> 8 images
    └───NORMAL/    -> 8 images
```



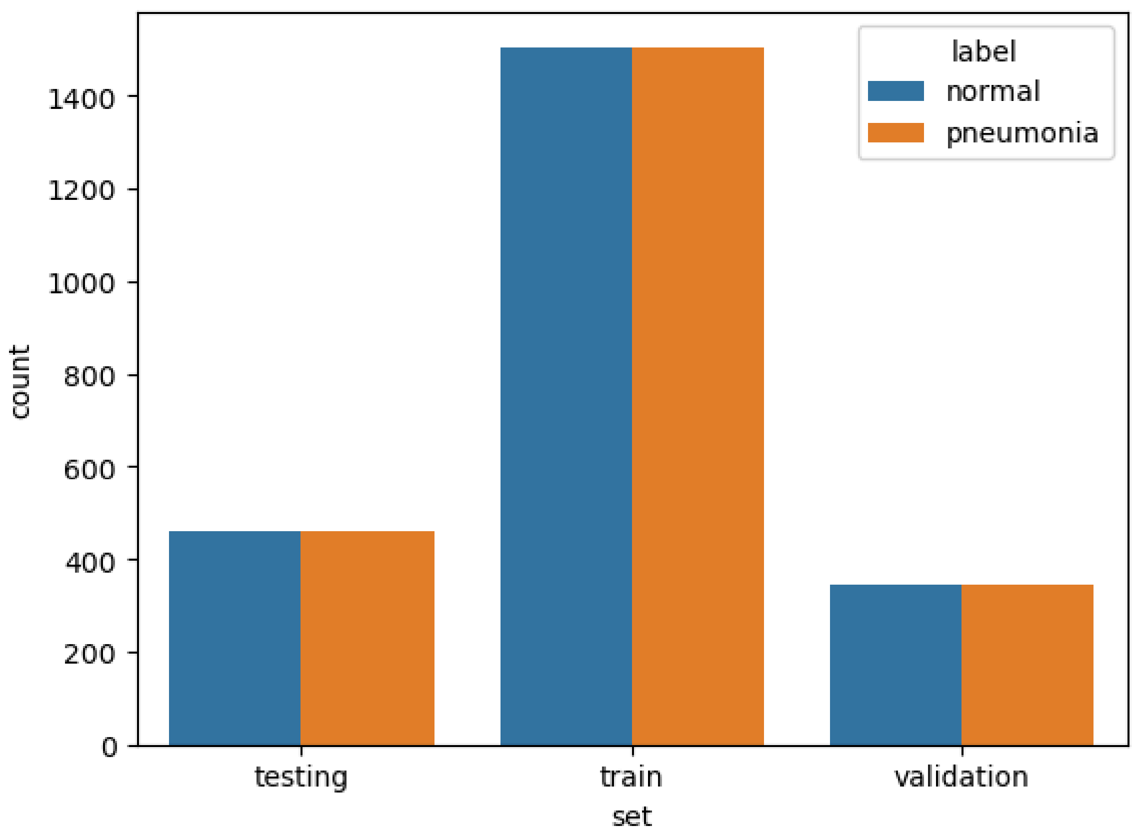

The dataset is highly imbalanced, which is going to impact the model's generalizability.

## Data Balancing and Splitting

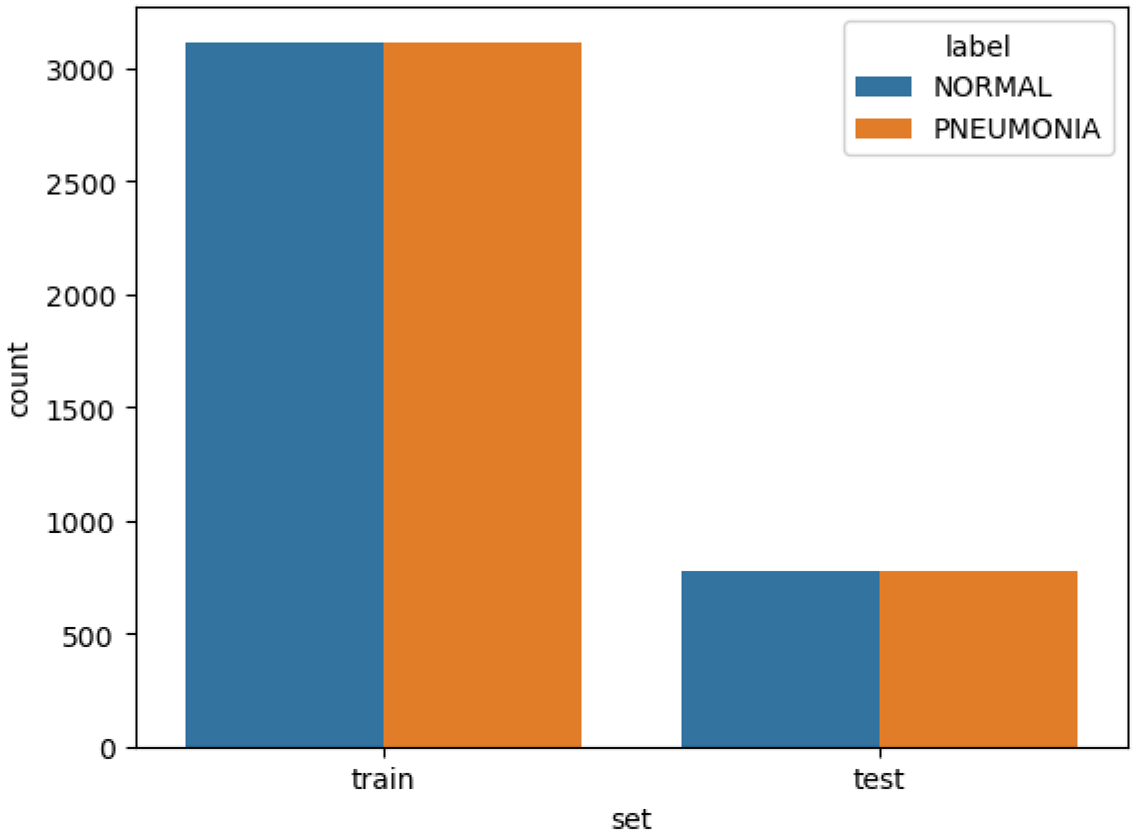
It was decided to add more images from another dataset from Kaggle (https://www.kaggle.com/datasets/salonimate/covid-pneumonia-nomal-xray):

```
New-CNP-Dataset/
├───testing/
│   ├───pneumonia/
│   ├───normal/
│   └───covid/
├───train/
│   ├───pneumonia/
│   ├───normal/
│   └───covid/
└───validation/
    ├───pneumonia/
    ├───normal/
    └───covid/
```

The second dataset is 1:1 balanced, however it has much less images for training. The plot of the number of images is shown below:



Preconfigured train/test splits in both datasets were reset. Datasets concatenated and shuffled. The combined dataset contained **3896 NORMAL** and **6586 PNEUMONIA** images, from which **3896 PNEUMONIA** images were randomly selected. The class distribution of the final dataset after **0.8/0.2 train/test split** turned out to be as follows:




## Image Augmentation

The final dataset was fed into the ImageDataGenerator for the Image Augmentation purposes. Various data augmentation techniques could be used in image processing, such as:

- Rotation
- Zooming
- Flipping
- Brightness and Contrast
- Blurring

Although most of these techniques may enhance the performance of image recognition models by introducing variations to the training images, *not all of them are useful for the medical field*, especially for applications like X-ray imaging.

For example, the human heart is almost always on the left (which appears on the right on a frontal chest X-ray), and flipping would give our model many instances of the rare condition dextrocardia (0.22%-1% incidence in the general population), which may skew the results.

Blurring is another example of a poor image augmentation technique for this task because a blurred X-ray is considered unusable and should be retaken.

In the end, we are left with two main categories of image augmentations:

- *Destructive* (those which remove image features and skew results)
- *Non-Destructive* (those which provide pixel variations needed to improve the model's performance)

In this project we used only non-destructive image manipulations:

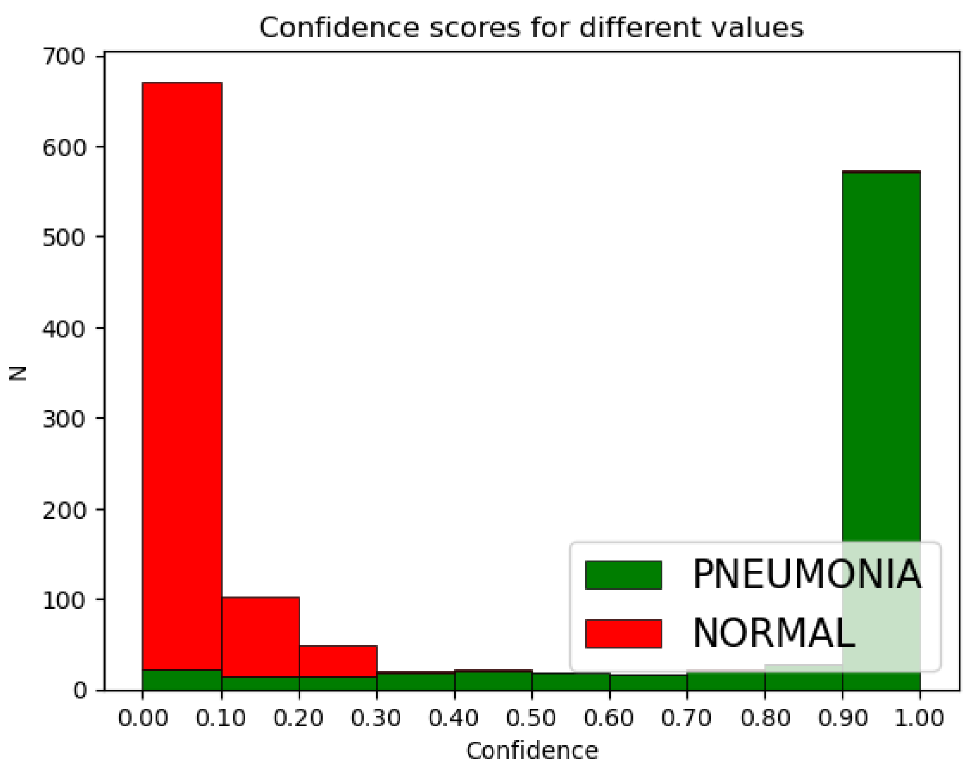
- Random Rotation: ±40°
- Random Width Shift: ±20%
- Random Height Shift: ±20%
- Random Shear: ±20%
- Random Zoom: ±20%

# Modeling - Transfer Learning

The convolutional neural network was built using the InceptionV3 with ImageNet weights. The input layer was reconfigured to accept RGB images 256x256 pixels in size. All pre-trained weights of the InceptionV3 model were frozen. The output layer was replaced with additional:
Flatten layer → Fully connected layer of 1024 neurons with ReLu activation function → Dropout(0.2) layer → Output layer of one node with Sigmoid activation function.
The neural network was compiled with Adam optimizer and Accuracy metric.
The model's training process showed gradual increase of model's accuracy, reaching 95%+ and decrease of model's loss, which is what we expect.

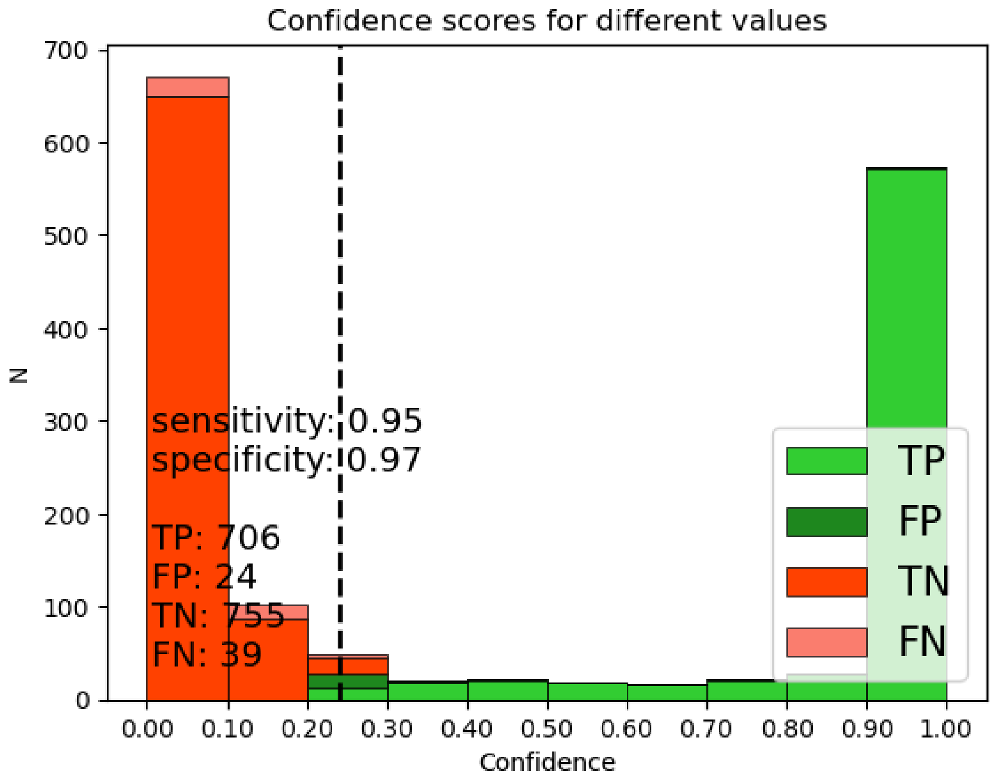
# Model's Performance Evaluation and Cutoff Tuning

The output layer of the neural network returns a number ranging from 0 to 1, which can be interpreted as "model's confidence" in its prediction, where 0 is NORMAL (negative) and 1 is PNEUMONIA (positive). However a good cutoff needs to be found in order to classify all outputs in the middle. For that all 1524 images from the test set were classified and the results were then plotted as a histogram with color-coded classes.

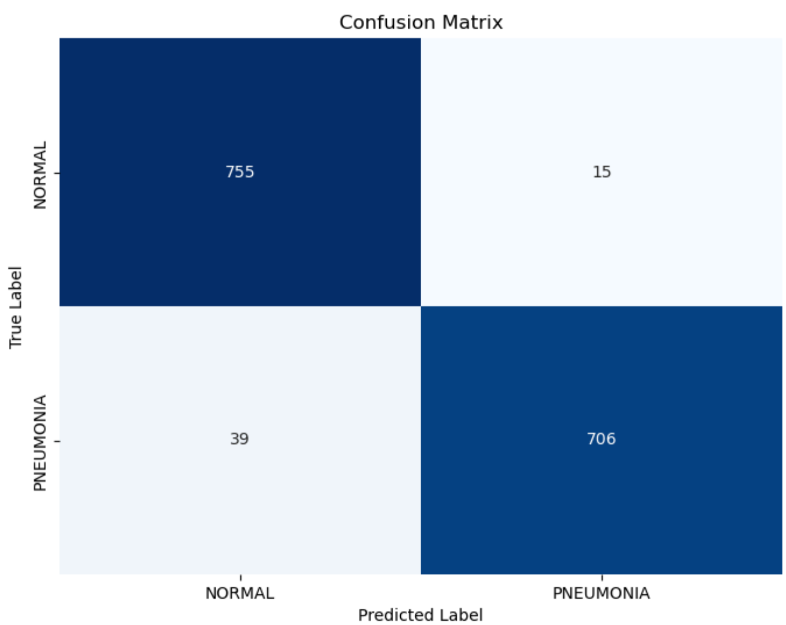



From the picture above it's clear that the cutoff point of 0.5 leaves too many pneumonia cases to the left of it. Also, we will need to account for True Positives, False Positives, True Negatives and

False Negatives. The next plot shows the save histogram which considers these metrics, and also, shows the chosen cutoff point and the metrics of the model's results, if this cutoff point is being used.



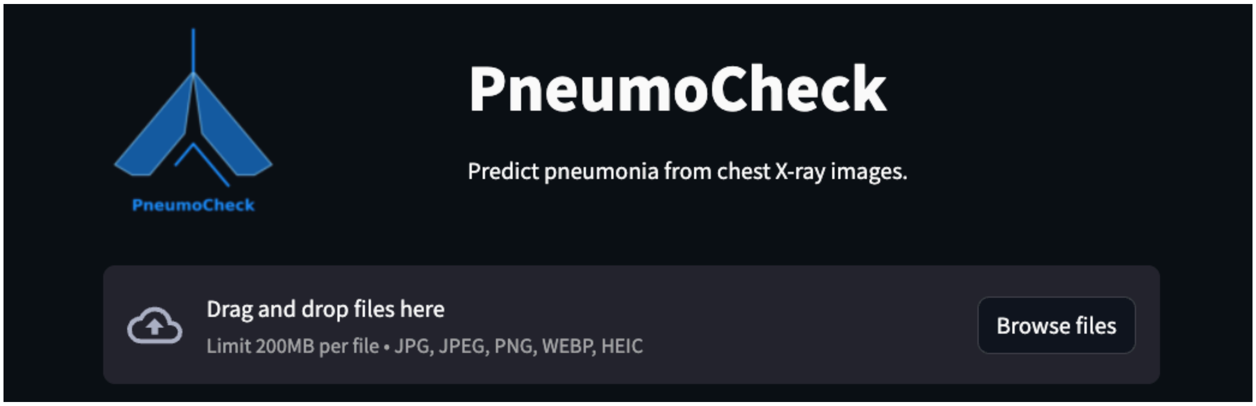

The best results (Sensitivity: 0.95, Specificity: 0.97) were achieved with the 0.24 cutoff, which is used in the final model.



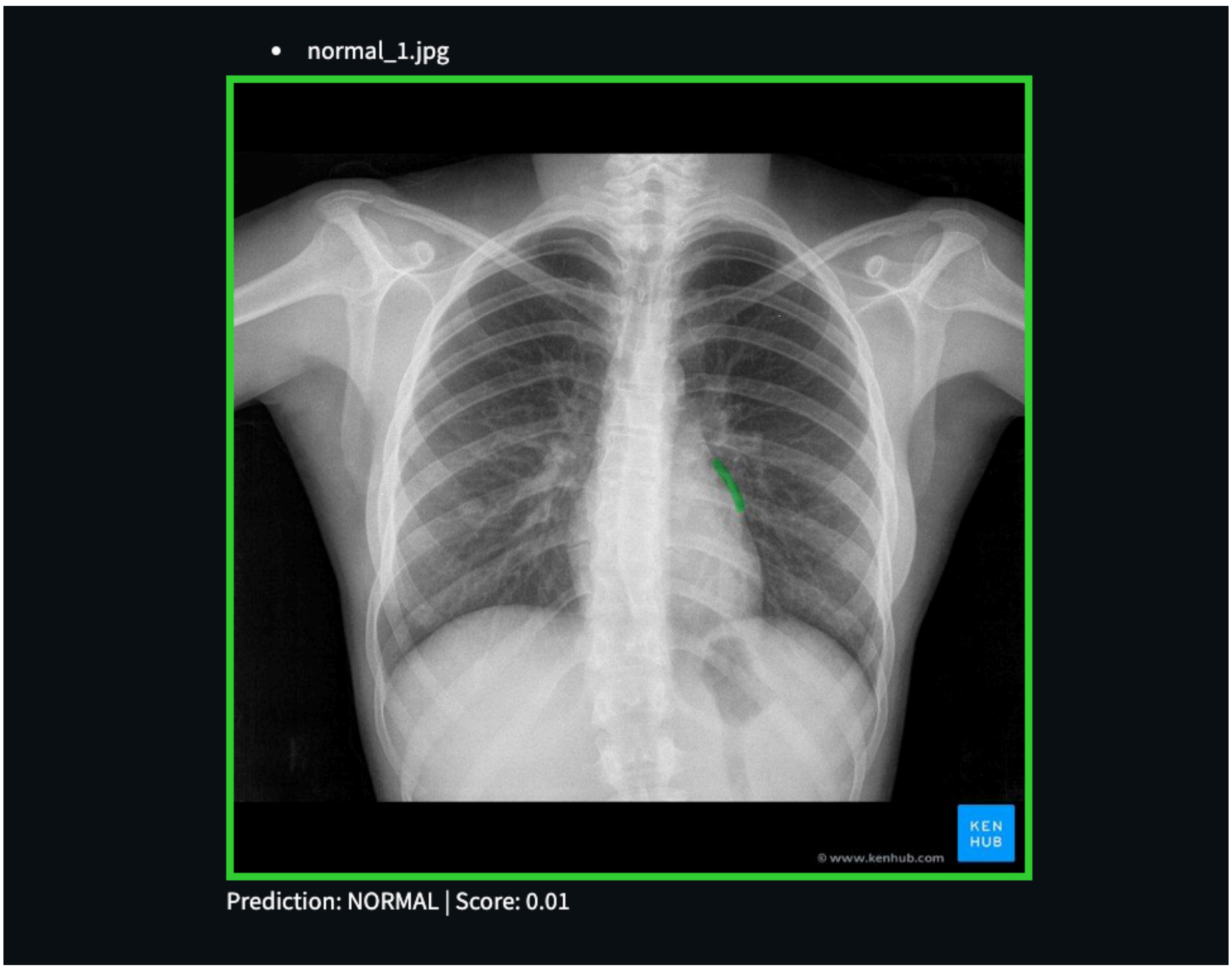

The confusion matrix looks very balanced and promising. The balanced inference results are very important because they show that the model has great generalizability when used to classify both NORMAL and PNEUMONIA classes.

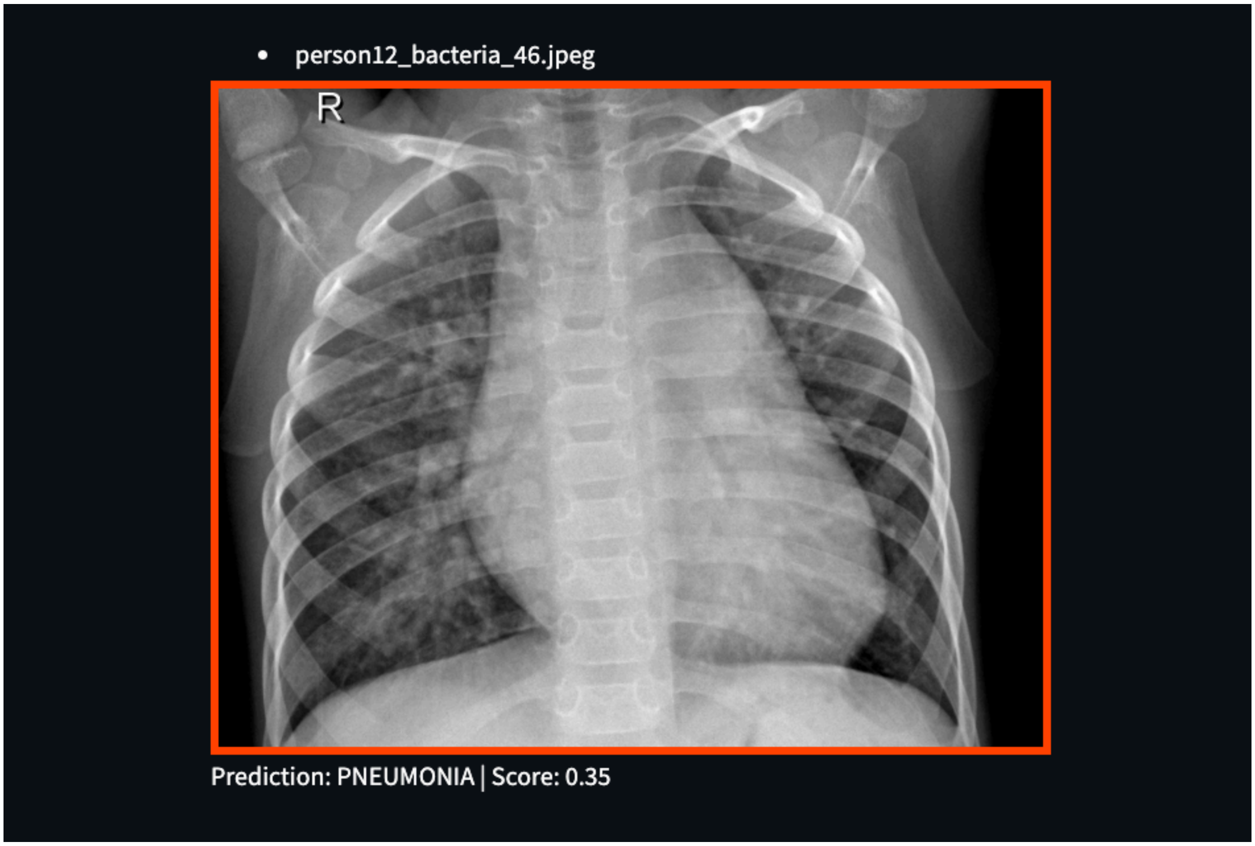
# Model Deployment

For the showcase the model is wrapped in a Web UI using the Streamlit library.




The user can upload images by dragging and dropping them into the browser window and get almost instant predictions.






All results are also stored in the table at the bottom of the page. It contains original file names, predicted class and model's confidence score. It can also be exported as a CSV file for further use.

| | File Name | Predicted Label | Score |
|---|---|---|---|
| 0 | normal_1.jpg | NORMAL | 0.0063 |
| 1 | normal_2.jpg | NORMAL | 0.0218 |
| 2 | normal_3.jpg | NORMAL | 0.0442 |
| 3 | normal_4.webp | NORMAL | 0.014 |
| 4 | normal_5.jpg | NORMAL | 0.0061 |
| 5 | normal_6.jpg | NORMAL | 0.0014 |
| 6 | normal_7.webp | NORMAL | 0.061 |
| 7 | normal_8.webp | NORMAL | 0.1248 |
| 8 | person12_bacteria_46.jpeg | PNEUMONIA | 0.351 |
| 9 | person19_bacteria_59.jpeg | PNEUMONIA | 0.506 |

Download predictions