International Journal of
Mathematics And its Applications
ISSN: 2347-1557

Int. J. Math. And Appl., **13(3)**(2025), 109–114
Available Online: http://ijmaa.in

Image Classification: Pneumonia Detection

Suvam Konar[1,*]

[1]*Thomas Worthington High School, 300 W Dublin-Granville Rd, Worthington, OH 43085, USA*

Abstract

This article examines the principles underlying supervised machine learning, convolutional neural networks, and transformation layers. These ideas were applied to a dataset of chest X-ray images, categorizing them into healthy and pneumonia classes. The accuracy of the model obtained is 85.3%. This article illustrates the power of machine learning as a diagnostic tool in medicine.

Keywords: supervised learning; Convolutional Neural Networks (CNNs); transformation layers; chest X-ray classification; pneumonia detection.

1. Introduction

Pneumonia is a lung infection caused by bacteria, viruses, or fungi that inflames the lungs, often filling them with fluid or pus and making breathing difficult. It has been recognized for centuries, and although treatments have improved greatly since the introduction of antibiotics in the 20th century, it still claims millions of lives each year, affecting both adults and children worldwide [1]. X-ray imaging is often used to confirm this diagnosis, as radiologists look for white or cloudy patches that indicate fluid buildup or infection [2]. Early diagnosis is especially important, as prompt treatment can significantly reduce the risk of severe complications or death, particularly in areas with limited healthcare access.

Machine learning allows computers to uncover complex patterns through data-driven models and deep learning [4,5]. These models can make predictions or decisions based on training data. Supervised learning, a subfield of machine learning, uses datasets where both inputs and outputs are labeled. In our case, the dataset consists of chest X-rays labeled as healthy or pneumonia. The model learns from this data and can then classify new X-rays, providing a probability of pneumonia for each patient. Machine Learning is the field of computer science that uses data to create computational models that can later be used to make predictions or decisions. By computational model, we mean an algorithm. Machine Learning and Statistics have a lot in common. The problem we study in this article belongs to the subfield of machine learning known as supervised learning.

*Corresponding author (konarsuvam@gmail.com)

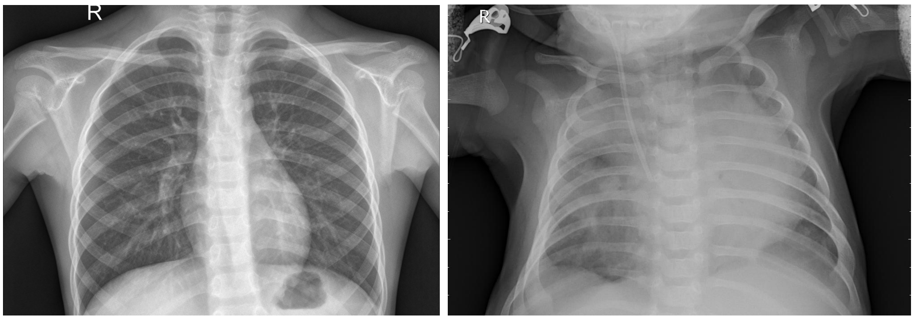

Figure 1: The left X-ray shows a healthy lung, while the right X-ray shows a pneumonia-infected lung. The lighter lower region in the pneumonia X-ray reflects fluid buildup. Images obtained from a publicly available dataset [3].

At a high level, supervised learning works by using a dataset where the inputs and outputs are known and labeled. In our case, the dataset consists of chest X-rays and the diagnosis of several thousand patients. The diagnosis is simply pneumonia or healthy. This dataset is known as the training set. It is important to stress that the diagnosis of all the patients in the training set is known. Machine learning techniques, which are explained in more detail in this article, are applied to the training, and as a result, we obtain an algorithm that is usually referred to as a model in the machine learning community. The model can be used as a diagnostic tool. More precisely, the model takes as input a patient's chest x-rays and provides as output the probability that the patient has pneumonia. A standard practice is to diagnose the patient as having pneumonia if the probability is larger than 0.5. Note that the model is usually used to diagnose patients who are not part of the training set. The main purpose of such models is to be used on patients whose diagnoses are unknown.

This paper will focus on image-based diagnostics. Here, our dataset is chest radiographs, where each picture is labeled with a known label indicating health status. We are training a convolutional neural network (CNN) on these labeled pairs to enable accurate classification of new, unseen X-rays of the same format. To ensure reliable performance, we split our dataset into distinct training, validation, and testing subsets, ensuring the model doesn't overfit to the training data and maintains good generalizability of all data. The model becomes a reliable pneumonia detector that can be applied to machines in hospitals and research, which supports faster and accurate treatments for patients.

2. Task

Our task is binary classification: each image belongs to either a "normal" or "pneumonia" category. We use one-hot encoding to convert labels into vectors: [1, 0] for normal, [0, 1] for pneumonia. The model outputs probabilities for both classes, and the predicted class is the one with the higher probability. The objective is to train the network such that predicted probabilities align with the true labels, enabling accurate classification of new images.

3. Neural Networks and Model Structure

At their core, machine learning models consist of numerous functions. Similar to mathematical functions, models take inputs such as numbers or images and then produce outputs such as predicted values or labels for their data. To make our comprehensive function, we use a type of machine learning model called a neural network, which consists of multiple layers of nodes or units, also known as neurons. Each of these nodes performs a simple calculation: it takes the input, multiplies it by internal weights, and adds a bias term. The result is then passed through what is known as an activation function, which essentially takes the input from the node and decides if the node should be 'fired'. This allows for models to identify non-linear trends within data and, as a result, classify more complex patterns.

The general structure of such networks is as follows:

- A sequence of layers forms a neural network.
- Our initial layer is the input, and the final layer is the output
- All layers between them are "hidden" layers, as the researcher does not interact with them
- Each layer has nodes that act as functions.
- The input of the input layer is the image data.
- The output of a layer is fed into the next layer.
- Our final layer outputs a prediction, which in our case will be 0 or 1, which corresponds to "normal" or "pneumonia."

The model in this paper was designed using the Keras Sequential API. A Python library that allows layers to be stacked in a linear order. Our model is based on a specialized type of model known as a convolutional neural network (CNN), which excels at identifying features of images. This works through the matrices that serve as layers or kernels that extract features from images. Convolutional layers include filters that slide over image pixels and detect local features. To further enhance our model, there are several types of layers. The ones we will use are convolutional, maxpooling, flattening, and dense layers. Briefly looking into each layer, Maxpooling gathers the highest values or most intense pixels, flattening which alters the input to be one-dimensional without losing information, and dense layers are fully connected, meaning a larger amount of connections between nodes and an ability to identify greater complexity in the data.

Outside of the network itself, when constructing a model, the factors we as researchers must consider are the type and number of layers, hyperparameters that affect each layer, and the type of activation function we would like to use.

4. Parameter Learning and Loss Minimization

Once the model architecture is defined, its behavior depends on two key categories of settings: parameters and hyperparameters. Parameters are the internal weights and biases of the network, which are learned automatically during training. Hyperparameters, on the other hand, are choices made by the researcher before training begins. They influence how learning happens, but they are not updated by the model itself. During training, the network starts with random weights and gradually updates them. For each training example, the model produces a prediction and compares it to the true label using categorical cross-entropy loss. The loss measures the distance between the prediction and the actual label. This process repeats for all training examples, gradually minimizing the average error across the dataset.

5. Input

The chest X-ray images used as inputs are preprocessed to ensure consistency and efficiency during training. Each image is represented as a 3-layer RGB matrix of shape 256×256×3, where the three layers correspond to red, green, and blue color intensities. Although X-rays are grayscale, they are stored in RGB format for compatibility with our model. Before training, all pixel values from 0 (black) to 255 (white) are normalized to the range [0, 1] by dividing by 255. This ensures that all features have the same scale, which speeds up learning and prevents large values from changing weight updates. The dataset is split into a training set (used to update weights), a validation set (used to monitor overfitting), and a test set (used to evaluate final performance on unseen data).

$$\begin{bmatrix} 255 & 0 & 0 & 0 & 255 \\ 255 & 255 & 0 & 255 & 0 \\ 255 & 255 & 0 & 255 & 255 \\ 255 & 0 & 0 & 255 & 255 \\ 255 & 0 & 255 & 255 & 255 \end{bmatrix}$$

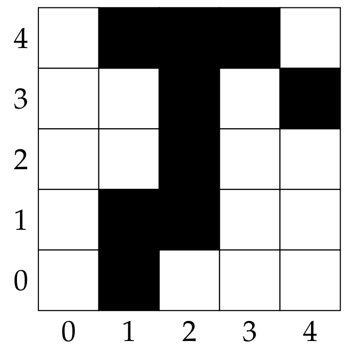

Figure 2: Example of the intensity in each pixel (left) and the resulting image (right).

6. Training and Evaluation

Training the model is done through showing the model examples and letting it adjust parameters to make better predictions. Each full pass through the training data is an epoch. At the end of each epoch, we check the performance of the model by checking the accuracy and loss using a separate group of images from the validation set. During training, we use an optimizer called Adam, which we will not go into the details here, but it helps the model change its internal values smartly and efficiently. We also choose a learning rate, which controls how big those changes are to prevent the model from too closely memorizing the training data, we use early stopping. This means we stop training when the model stops improving on the validation set, even if we haven't reached the maximum number of epochs. Another method we use is training in small batches instead of feeding the entire dataset at once. This makes training more stable and can help the model find better solutions. Over time, the model learns which filters and patterns are most useful for recognizing signs of pneumonia in the lungs to evaluate our model, the key metrics we use are accuracy (how often the model gets the right answer) and loss (how uncertain the predictions are). These results help us understand how well the model would apply in real-world situations where new data is input.

7. Applications to Pneumonia X-rays

Our dataset consisted of over 5000 images used for both training and testing. While we will not go into detail regarding the exact meaning of every hyperparameter in each layer, we can report the architecture that performed best after testing several different configurations.
The model architecture is as follows:

- First convolutional layer: 32 filters, ReLU activation.
- Max pooling layer to reduce feature size.
- Flattening layer.
- Dense layer: 32 nodes, ReLU activation, L2 regularization.
- Dense layer: 8 nodes, ReLU activation, L2 regularization.
- Output layer: 2 nodes, softmax activation.

8. Conclusions

Machine learning is a powerful field of computer science with applications in numerous fields. Convolutional neural networks are a powerful tool for image-based diagnostics. This study illustrates that CNNs can reliably classify chest X-rays into healthy and pneumonia categories, supporting faster and more accurate medical diagnoses.

References

[1] W. S. Lim, *Pneumonia—Overview*, Encyclopedia of Respiratory Medicine, Elsevier, (2022), 185–197.

[2] R. Kundu, R. Das, Z. W. Geem, G. T. Han and R. Sarkar, *Pneumonia detection in chest X-ray images using an ensemble of deep learning models*, PLOS One, (2021).

[3] D. S. Kermany, K. Zhang and M. Goldbaum, *Labeled Optical Coherence Tomography (OCT) and Chest X-Ray Images for Classification*, Kaggle, (2018).

[4] M. P. Deisenroth, A. A. Faisal and C. S. Ong, *Mathematics for Machine Learning*, Cambridge University Press, (2020).

[5] A. Karpathy, F. -F. Li and J. Johnson, *Convolutional neural networks for visual recognition*, Stanford University, (2024).