Detecting Pneumonia from Chest X-Ray Images Using Convolutional Neural Networks

Brandon Samson
California State University, Long Beach
brandon.samson01@student.csulb.edu

Bryan Tran
California State University, Long Beach
bryan.tran03@student.csulb.edu

December 17, 2025
Github: https://github.com/brandon-smsn/CECS-456-Project

Abstract

This report presents a comparative study of two convolutional neural network (CNN) models for binary classification of chest X-ray images into NORMAL and PNEUMONIA categories. Using data augmentation and proper regularization, both models were trained on the same dataset and evaluated on a shared test set. Performance metrics including accuracy, precision, recall, and F1-score are reported, and insights are drawn by comparing the two models.

1. Introduction

1.1 Background

Pneumonia is an infection of the lungs that causes fluid in the lungs and leads to symptoms such as shortness of breath, chills, fever and severe cough. Currently, pneumonia can be diagnosed through chest x-rays by displaying white patches called infiltrates. These x-rays help differentiate pneumonia from other illnesses like bronchitis and can reveal its severity. They are essential for diagnosing pneumonia because it's cost- effective, fast, and can provide a more accurate diagnosis than physical exams.

1.2 Task

Images of chest x-rays can be classified between two groups: normal and pneumonia. By using a convolutional neural network or (CNN) on a dataset of chest x-rays images, we can train a model that determines whether an image of a chest x-ray shows signs of pneumonia.

2. Dataset and Related Work

2.1 Dataset

The dataset we will be using is the Chest X-Ray Images (Pneumonia) from Kaggle. There are 5,216 training images, 16 validation images, and 624 test images. There is a binary class of (0) NORMAL and (1) PNEUMONIA. Our preprocessing will have pixel values rescaled to [0,1] along with our data augmentation applied to the training set which includes: rotation, shifts, shear, zoom, and horizontal flips.

2.2 Related Work

CNNs have been used successfully in the medical field to classify images such as pneumonia, tuberculosis, and diabetes. Many models have been created using a multitude of convolutional blocks, batch normalization, pooling layers, and dropouts to prevent overfitting. CNNs are tools that guide the medical field into quick analysis for patients, providing an accurate and efficient method to incorporate in a possibly life and death situation.

3. Methodology

3.1 Data Augmentation

To prevent overfitting and enhance model generalization, both models were trained using the same augmentation strategy. This included:

- Rotation: ±15°
- Width/Height shift: ±10%
- Shear and zoom: ±10%
- Horizontal flip
- Fill mode: nearest

This helps us have similar data to be used for our models and helps them generalize better by simulating variations that may arise in the real-world.

3.2 Model 1 Architecture

Model 1 is a sequential CNN with 3 convolutional blocks that takes 150x150x3 RGB images as input. Each block has 2 layers with batch normalization, MaxPooling and dropout (0.25). There are two dense layers of 256 units and 128 units with dropout (0.5). This model is designed to prevent overfitting throughout all layers.

3.3 Model 2 Architecture

Model 2 consists of four convolutional blocks with a hybrid approach of MaxPooling in the early layers and AveragePooling in the later layers. Each block included batch normalization and progressively increasing dropout (0.1-0.25). A single dense layer of 256 units with dropout (0.5) preceded the sigmoid output layer. Overall, this architecture was designed to preserve features in early layers while regularizing deeper layers to improve generalization.

3.4 Loss Function & Metrics

Both models were trained using the Adam optimizer and binary cross-entropy loss, due to the nature of having only two classes. Performance was evaluated using metrics such as accuracy, precision, recall, and F1-score to ensure reliability in detecting positive pneumonia cases, especially in the use case and importance of minimizing false negatives.

4. Experimental Setup

Tools used included the dataset from Kaggle, VSCode with a virtual Python environment, TensorFlow v2.20.0. Our Models were trained with a batch size of 32 for a maximum of 25 epochs. Model 1 resolved at the 16th epoch, while Model 2 resolved on the 18th. Both models used ReduceLROOnPlateau as a way to lower the learning rate when stuck while EarlyStopping monitored validation loss if it didn't improve. Callbacks were employed to save the best-performing model during training and to dynamically adjust the learning rate.

5. Measurements

Multiple metrics were utilized to evaluate each model's performance. These include accuracy, precision, recall, f1-score, loss curves, and confusion matrix. For accuracy, we calculated it through the proportion of correct predictions and total predictions. For precisions we used the proportion of correct pneumonia predictions and all pneumonia predictions. We placed extra importance on precision as it means fewer healthy patients are misdiagnosed with pneumonia. Recall was calculated by the proportion of actual pneumonia cases correctly identified. F1-score was calculated through the harmonic mean of precision and recall. Loss curves were measured to track training and validation loss to monitor overfitting. Finally, a confusion matrix was used to create a visual representation of prediction vs. actual.

6. Results Analysis, Intuitions, and Comparison

6.1 Model 1 Performance

Model 1 achieved a test accuracy of 89.58%, with a precision of 0.8916, recall of 0.9487, and an F1-score of 0.9193. The loss on the test set was 0.2863. The high recall indicates effective detection of pneumonia cases, though slight overfitting was observed due to fluctuations in validation performance.

Figure 1: Model 1 Confusion Matrix

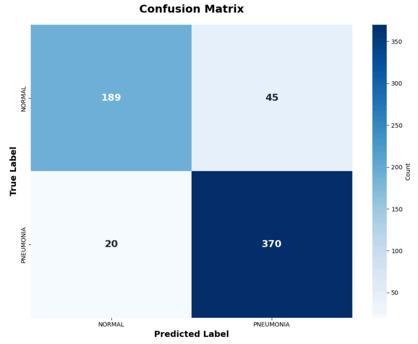


Figure 2: Model 1 Accuracy Graph

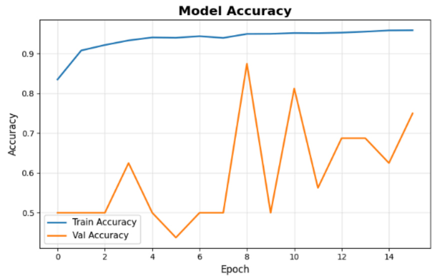
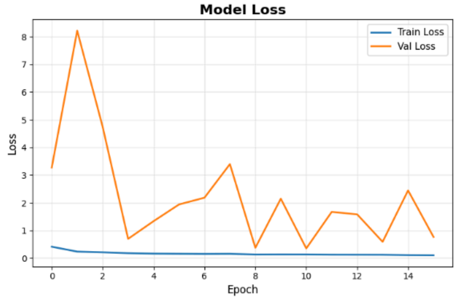


Figure 3: Model 1 Loss Graph



6.2 Model 2 Performance

Model 2 achieved slightly higher performance, with a test accuracy of 90.38%, precision of 0.8966, recall of 0.9564, and an F1-score of 0.9256. The test loss was 0.2598. The use of mixed pooling layers and progressively increasing dropout contributed to improved generalization and slightly better detection of positive cases.

Figure 4: Model 2 Confusion Matrix

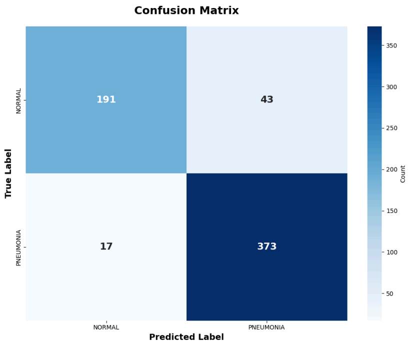


Figure 5: Model 2 Accuracy Graph

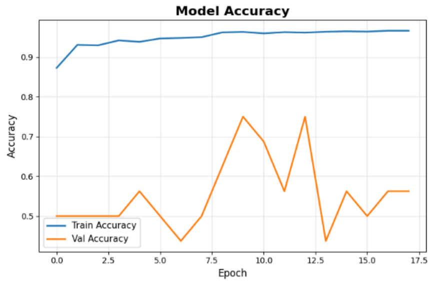
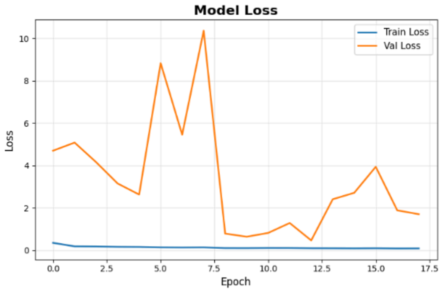


Figure 6: Model 2 Loss Graph



6.3 Comparison and Insight

6.3.1 Model 1 vs. Model 2

Both models were CNN trained on the same dataset but have differences in architecture and regularization approaches. Model 1 emphasized on consistent regularization while Model 2 uses hybrid pooling techniques and progressive dropout strategies. Model 2 had a slightly better performance in accuracy than Model 1 with an accuracy of 90.38% versus Model 1 with an accuracy of 85.58%. Model 2 also had a higher recall signifying a better detection of pneumonia diagnosis. Model 2 performed better because of an additional convolutional and a hybrid pooling strategy that preserves critical features in early layers while smoothing representations in deeper layers. However, Model 1's simple architecture has the advantages of computational efficiency and training speed. Fewer parameters also means reducing the risk of overfitting on a limited set of imaging data.

6.3.2 Model 1 Graph Analysis

In Figure 2, the validation accuracy peaks at epoch 8, declines at epoch 9, and increases again at epoch 10. This creates a zig-zag pattern that is caused by a small validation set (16 images) from the dataset. The erratic behavior of the validation accuracy and the smooth training curve shows an issue of overfitting and insufficient validation data.

6.3.3 Model 2 Graph Analysis

In Figure 5, we can see that validation accuracy peaks at epoch 7 before declining while training accuracy improved throughout. This indicates an overfitting problem despite the use of regularization. Again, the small validation set is what contributed to these results which is an issue of the dataset that we used.

7. Conclusion

This study developed two CNN models for the purpose of detecting pneumonia from chest x-ray images. Model 1 achieved an 85.58% accuracy and Model 2 had a 90.38% accuracy. The performance increase from Model 1 to Model 2 was the result of an increase in convolutional blocks, a hybrid pooling strategy, and progressive dropout regularization. In both models, overfitting occurred because of a small validation set. This study demonstrated the impact of architectural depth and regularization in improving image classification. Despite the limitations of the dataset, this project shows how CNNs can be applied for automatic pneumonia examinations.

8. Contributions

Brandon developed, designed, and implemented Model 1. He also wrote the introduction, description of Model 1's architecture, the types of measurements, the comparison of Model 1 and Model 2, the graph analysis, and the conclusion in this report. Bryan developed, designed, and implemented Model 2. He wrote about the dataset, methodology, experimental setup, each model's performance, and included all charts and images.