Pneumonia Classification Using ResNet-18

Zahra Khodagholi

March 16, 2024

1 Task 1.1: Training a Model from Scratch

1.1 Introduction

This report outlines the development and performance of a deep learning model for the classification of chest X-ray images into two categories: Normal and Pneumonia. The model is based on the ResNet-18 architecture, trained from scratch on a labeled dataset.

1.2 Implementation Details

1.2.1 Network Architecture

The model employed is ResNet-18, modified to output two classes instead of the original 1000 classes designed for ImageNet. The final fully connected layer was replaced with a new one having two output features.

1.2.2 Data Preprocessing

Images were resized to 224×224 pixels, converted to tensors, and normalized using mean values of $[0.485, 0.456, 0.406]$ and standard deviation values of $[0.229, 0.224, 0.225]$.

1.2.3 Hyperparameters

- Learning Rate: 0.001
- Momentum: 0.9
- Batch Size: 32
- Number of Epochs: 20

1.2.4 Optimization

Stochastic Gradient Descent (SGD) was used as the optimizer.

1.2.5 Loss Function

Cross Entropy Loss was utilized to quantify the difference between predicted and true labels.

1.3 Results

1.3.1 Training and Validation Loss Curves

Figure 3 shows the training and validation loss over 20 epochs. The graphs indicate [provide your observations here].

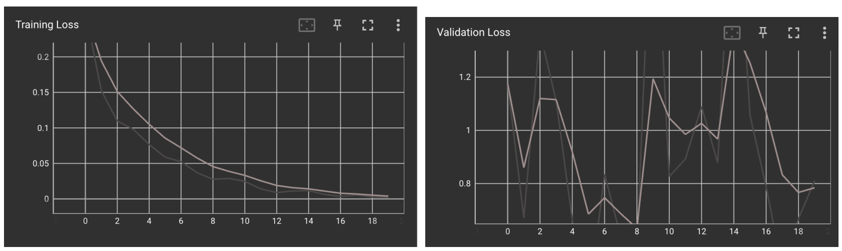

Figure 1: Training and Validation Loss Curves.

1.3.2 Classification Accuracy

Overall Test Set Accuracy The model achieved an overall accuracy of 75.48% on the test set.

Class-specific Accuracy

- Normal: 35.04%
- Pneumonia: 99.74%

1.3.3 Misclassified Images

Figure **??** showcases several misclassified images, providing insight into cases where the model failed to predict accurately.

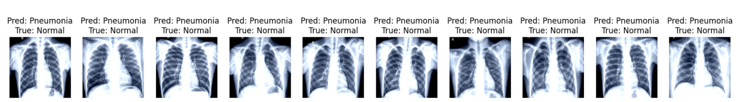

Figure 2: Examples of Misclassified Images.

2 Task 1.2: Fine-Tuning a Pre-trained Model

For Task 1.2, we fine-tuned a pre-trained ResNet-18 model on the chest X-ray dataset. The model, initially trained on the ImageNet dataset, was adapted for binary classification by replacing the final fully connected layer.

2.1 Implementation Details

- **Model Architecture:** ResNet-18 pre-trained on ImageNet, with the final layer adjusted to output two classes.
- **Data Augmentation:** Training data transformations included resizing to 224x224 pixels, random horizontal flipping, conversion to tensor, and normalization. Validation data were only resized, converted, and normalized without flipping.
- **Loss Function:** Cross Entropy Loss, suitable for classification tasks.
- **Optimizer:** Stochastic Gradient Descent (SGD) with a learning rate of 0.001 and momentum of 0.9, applied only to the parameters of the final layer to fine-tune the pre-trained model.
- **Batch Size:** 32 images per batch.
- **Epochs:** The model was trained for 20 epochs.
- **Evaluation Metrics:** Model performance was evaluated using accuracy, both overall and per class (Normal, Pneumonia).

2.2 Results

The training process was monitored using TensorBoard, which provided valuable insights into the loss and accuracy trends over epochs.

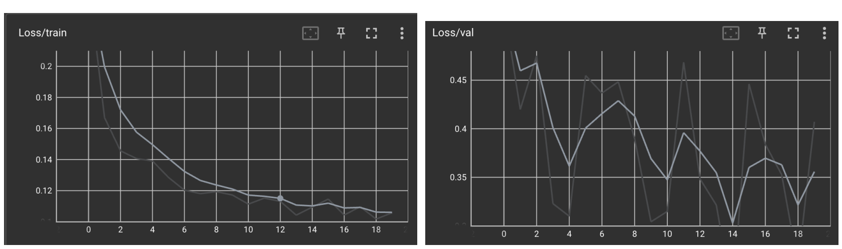

Figure 3: Training and Validation Loss Curves.

2.2.1 Evaluation on Test Set

The fine-tuned model was evaluated on a separate test set, resulting in an overall test accuracy of 80.13%. The accuracy for detecting 'Normal' cases was 50.00%, and for 'Pneumonia' cases, it was 98.21%.

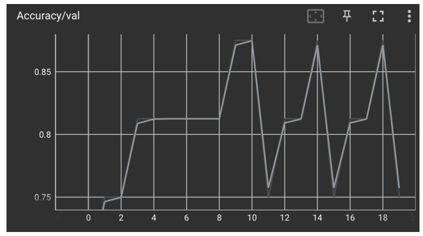

Figure 4: Validation Accuracy Curve for Fine-Tuned Model

2.3 Failure Case Analysis

To gain further insights into the model's performance, we analyzed several misclassified images from the test set. These cases help identify potential limitations and areas for improvement.

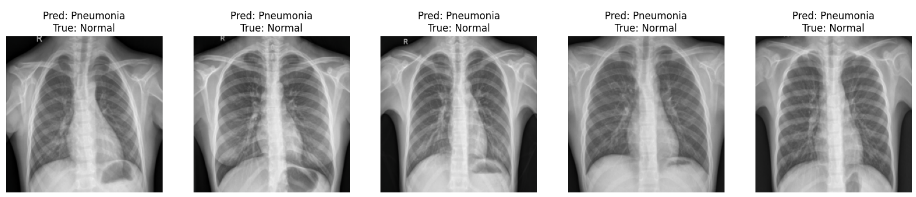

Figure 5: Misclassified Images by the Fine-Tuned Model

2.4 Discussion and Future Work

The fine-tuning approach demonstrated significant promise, leveraging pre-trained features from ImageNet to effectively classify chest X-ray images. However, there are areas for enhancement:

- **Advanced Data Augmentation:** Incorporating more sophisticated augmentation techniques could improve model robustness.
- **Extended Hyperparameter Tuning:** Further optimization of the learning rate, batch size, and other hyperparameters could yield better results.
- **Deeper Architectures:** Exploring more complex models like ResNet-50 or DenseNet could capture more detailed features relevant to the task.
- **Class Imbalance Handling:** Implementing strategies to mitigate class imbalance, such as weighted loss functions, could improve model fairness.

These potential improvements form the basis for future work, aiming to enhance the model's accuracy and reliability in medical diagnostic applications.