Chest X-Ray Pneumonia Classification Project

Task 1: First Steps

When I first received the Chest X-Ray Pneumonia dataset, I took these essential steps:

Initial Analysis

- Examined the folder structure (train, validation, test folders with NORMAL/PNEUMONIA subfolders)
- Counted images in each category to understand the dataset size
- Looked at sample images to understand their characteristics

Key Observations

- **Class Imbalance**: Found 3106 pneumonia images vs. only 1079 normal images in the training set (almost 3:1 ratio)
- **Small Validation Set**: Only 16 images in the original validation folder, too few for reliable evaluation
- **Image Variations**: X-rays differ in brightness, contrast, and positioning
- **Quality Issues**: Some images have text markers or poor contrast

```
# Sample code for initial class distribution analysis
normal_count = len([f for f in train_files if "NORMAL" in f])
pneumonia_count = len([f for f in train_files if "PNEUMONIA" in f])
print(f"Training images: Normal = {normal_count}, Pneumonia =
{pneumonia_count}")
```

Task 2: Data Preprocessing

I implemented these preprocessing steps to prepare the data:

Step	What I Did	Why It's Important
Dataset Restructuring	Combined train and validation sets, then split into 80:20 ratio (4185 train, 1047 validation)	Original validation set was too small (16 images)
Image Resizing	Standardized all images to 150×150 pixels	Neural networks require consistent input dimensions
Data Augmentation	Added horizontal flips, slight rotations (±10°), small position shifts	Creates more training examples and improves model robustness
Brightness/Contrast Adjustment	Varied brightness and contrast by ±20%	X-ray machines produce images with different exposure settings
Normalization	Scaled pixel values using ImageNet statistics	Helps the model train faster and improves transfer learning
Class Imbalance Handling	Applied class weights (Normal: 1.94, Pneumonia: 0.67) and Focal Loss	Prevents bias toward the majority class (pneumonia)

All validation and test images underwent only resizing and normalization (no augmentation) to ensure fair evaluation.

Task 3: Model Selection

My Choice: Custom CNN Architecture

I designed a custom CNN with this structure:

- 5 convolutional blocks with increasing feature maps (32→64→128→256→512)
- Each block contains two convolutional layers with batch normalization
- Dropout layers to prevent overfitting
- Final dense layers that narrow down to a single output (binary classification)

Why I Chose This Model:

1. **Medical Image Specific**:
    - X-rays contain subtle patterns that CNNs excel at detecting
    - The progressive strcuture captures features at multiple scales (from small edges to larger patterns)
2. **Practical Considerations**:
    - Complex enough to learn important patterns but not so complex that it would overfit
    - Can train on standard hardware in reasonable time
    - Balanced for the available dataset size (~5,000 images)
3. **Design Rationale**:
    - Double convolution blocks capture more complex features
    - Batch normalization makes training more stable with medical images
    - Dropout prevents the model from memorizing the training data

This design creates a good baseline that's specifically tailored for chest X-ray analysis.

Task 4: Metrics Selection

For pneumonia classification, I chose these evaluation metrics:

Metric	Why It's Important for Pneumonia Detection
Accuracy	Basic measure of correct predictions, but can be misleading with imbalanced data.
Precision	Measures how many "pneumonia" predictions are actually pneumonia. Important to avoid unnecessary treatments.
Recall	Measures how many actual pneumonia cases the model finds. Critical because missing pneumonia can be life-threatening.
AUROC	Evaluates model performance across different thresholds, useful for finding optimal clinical settings.
F1-Score	Balances precision and recall, providing a single metric when both false positives and false negatives matter.

Clinical Priority: In pneumonia detection, recall (sensitivity) is especially important because missing a case of pneumonia (false negative) could lead to delayed treatment and serious health consequences. However, precision also matters to avoid unnecessary antibiotic use and healthcare costs.

Task 5: Experiment Roadmap

I planned these 10 experiments to systematically improve the model:

1. **Baseline Model** - Train the initial custom CNN with standard settings to establish performance benchmarks
2. **Loss Function Comparison** - Compare regular BCE loss vs. Focal Loss to address class imbalance
3. **Augmentation Testing** - Determine which image augmentations (flips, rotations, brightness) help most
4. **Learning Rate Stud** - Test different learning rates (1e-5 to 1e-2) to find the optimal training speed
5. **Batch Size Optimization** - Compare batch sizes (8, 16, 32, 64) for best training stability and performance
6. **Dropout Tuning** - Test different dropout rates to find the best balance between learning and overfitting
7. **Network Depth/Width** - Try adding or removing layers to see if a deeper or wider network helps
8. **Model Ensemble** - Combine predictions from multiple model variations to improve overall performance
9. **Transfer Learning** - Apply pre-trained ResNet50 model to leverage knowledge from other image datasets
10. **Fine-Tuning Strategy** - Test different approaches to freezing/unfreezing layers in the transfer learning model

Each experiment builds on previous findings, allowing systematic improvement while understanding which changes actually help.

Task 6: Transfer Learning

Implementation

After training my custom CNN, I implemented transfer learning using ResNet50 pre-trained on ImageNet:

```
# Load pre-trained model
pretrained_model = models.resnet50(pretrained=True)

# Replace the final classification layer
pretrained_model.fc = nn.Sequential(
    nn.Linear(2048, 512),
    nn.BatchNorm1d(512),
    nn.ReLU(),
    nn.Dropout(0.4),
    nn.Linear(512, 128),
    nn.ReLU(),
    nn.Linear(128, 1),
    nn.Sigmoid()
)

# Freeze early layers to preserve learned features
for param in list(pretrained_model.parameters())[:-36]:
    param.requires_grad = False
```

Performance Comparison

Metric	Custom CNN	Transfer Learning	Improvement
Accuracy	78.0%	82.9%	+4.9%
Precision	74.1%	78.8%	+4.7%
Recall	99.7%	99.2%	-0.5%
AUROC	94.8%	96.0%	+1.2%

Added Value Assessment

Transfer learning provided these benefits:

1. **Better Performance** - Improved almost all metrics, especially accuracy and precision
2. **Faster Learning** - Reached good performance in fewer epochs (10 vs. 15)
3. **Better Classification of Normal Cases** - Normal recall improved from 41.9% to 55.6% (+13.7%)
4. **Efficiency** - Despite more parameters, training was similar in time due to faster convergence

5. **Clinical Impact** - The improved precision means fewer false alarms, while still maintaining the critical high recall for pneumonia

The transfer learning approach proved superior for this task, showing that knowledge from natural images transfers well to chest X-rays despite the domain difference.

Task 7: Model Interpretability

To understand what features the model uses to make predictions, I implemented Gradient-weighted Class Activation Mapping (GradCAM):

Implementation Approach

GradCAM works by:

1. Capturing activations from a deep convolutional layer
2. Computing gradients of the prediction with respect to these activations
3. Weighting activation maps based on their importance to the prediction
4. Creating a heatmap showing regions that influenced the decision

```
def simple_gradcam(model, image, target_layer):
    # Capture activations and gradients
    # Weight activations by gradients
    # Generate heatmap highlighting important regions
    # (Simplified version of the full implementation)
```

Key Findings

By analyzing GradCAM visualizations:

1. **Correct Pneumonia Predictions**: The model focuses on areas of lung opacity and consolidation, matching radiologists' focus areas
2. **Correct Normal Predictions**: The model shows more diffuse attention across clear lung fields
3. **Error Analysis**: False positives often occur when the model focuses on normal anatomical structures like the heart border or bone overlaps
4. **Model Comparison**: The transfer learning model shows more precise focus on medically relevant areas compared to the custom CNN
5. **Quantitative Assessment**: Analysis showed the transfer learning model was more likely to focus on actual lung regions rather than artifacts or borders

Clinical Value of Explainability

These visualizations:

- Build trust by showing the model looks at medically relevant areas
- Help identify when the model might be using shortcuts or artifacts
- Could assist radiologists by highlighting suspicious areas
- Provide a teaching tool for medical students learning to identify pneumonia

Conclusion: From Images to Insights

This pneumonia classification project demonstrates how deep learning can effectively analyze chest X-rays to detect pneumonia with high accuracy while providing explainable results.

Key Achievements:

- Developed two effective models for pneumonia detction with complementary strengths
- Successfully applied transfer learning to improve performance across most metrics
- Implemented robust data preprocessing to handle real-world variations in medical images
- Created visual explanations that make the model's decisions transparent and trustworthy

Clinical Relevance:

Both models achieved exceptional sensitivity for pneumonia detection:

- Custom CNN: 99.7% recall for pneumonia cases
- Transfer Learning model: 99.2% recall for pneumonia cases

This means both models would rarely miss a pneumonia case, which is the primary clinical concern. However, the transfer learning model showed better discrimination of normal cases (55.6% vs 41.9% recall for normal), reducing false positives.

Class Imbalance Insights:

Our results reveal an important challenge in medical image classification:

- Both models excelled at detecting the majority class (pneumonia)
- Both struggled more with correctly classifying the minority class (normal)
- The transfer learning approach significantly improved normal case detection

Model Selection Insights:

While both models performed well, the transfer learning approach demonstrated clear advantages:

1. Higher overall accuracy (82.9% vs 78.0%)
2. Better precision for pneumonia detection (78.8% vs 74.1%)
3. More balanced performance across both classes
4. Better F1-scores for both normal (0.71 vs 0.59) and pneumonia (0.88 vs 0.85) classes

These findings suggest that knowledge from natural image datasets (ImageNet) transfers surprisingly well to medical domains when properly implemented.

Limitations and Future Work:

- The dataset size (~5,000 images) is relatively small by deep learning standards
- Both models still produce false positives on normal cases that could lead to unnecessary treatments
- External validation on data from different hospitals and imaging equipment would strengthen reliability
- Integration with clinical metadata (patient age, symptoms, lab values) could further improve performance

Final Thoughts:

This project demonstrates the potential of AI to assist in medical diagnosis while maintaining transparency through explainable AI techniques. The combination of high sensitivity and model interpretability addresses two critical requirements for AI in healthcare: effectiveness and trustworthiness.

The transfer learning model's improved performance on normal cases while maintaining excellent pneumonia detection makes it the preferred approach for potential clinical implementation. However, the remaining challenge of false positives (normal X-rays classified as pneumonia) suggests that these models would be best deployed as assistive tools for radiologists rather than autonomous diagnostic systems.