



Detecção de Tuberculose em Radiografias de Tórax

Utilizando Deep Learning e Transfer Learning

Evandro Rocha

Universidade Federal do Rio de Janeiro
UFRJ/COPPE/PEE

December 9, 2025



Agenda



Introdução



Tuberculose: Um Problema Global

- ▶ 10 milhões de casos/ano (OMS, 2022)
- ▶ 1,5 milhões de mortes/ano
- ▶ Principal causa de morte por doença infecciosa
- ▶ Diagnóstico precoce é crucial

Desafios do Diagnóstico

- ▶ Escassez de radiologistas
- ▶ Variabilidade inter-observador
- ▶ Tempo de análise
- ▶ Custo elevado



Motivação

Por que Deep Learning?

- ▶ Capacidade de aprender características complexas
- ▶ Alto desempenho em tarefas de visão computacional
- ▶ Potencial para auxiliar diagnóstico médico
- ▶ Redução de tempo e custo

Objetivo do Trabalho

Desenvolver e avaliar modelos de Deep Learning para detecção automática de tuberculose em radiografias de tórax do dataset Shenzhen Hospital.



Problema Abordado



Definição do Problema

Problema de Classificação Binária

Dado uma radiografia de tórax (raio-X), classificar como:

- ▶ **Classe 0:** Normal (sem tuberculose)
- ▶ **Classe 1:** Tuberculose (presença da doença)

Desafios Técnicos:

- ▶ Variabilidade nas imagens
- ▶ Sutileza dos padrões
- ▶ Dataset limitado (662 imagens)
- ▶ Desbalanceamento de classes

Requisitos:

- ▶ Alta sensibilidade (recall)
- ▶ Alta especificidade
- ▶ Interpretabilidade
- ▶ Eficiência computacional



Dataset Shenzhen Hospital

Características do Dataset:

- ▶ **Total:** 662 radiografias
- ▶ **Normal:** 326 imagens
- ▶ **Tuberculose:** 336 imagens
- ▶ **Formato:** PNG, grayscale
- ▶ **Resolução:** Variável
- ▶ **Fonte:** NIH/NLM

Divisão dos Dados:

Conjunto	%
Treino	70%
Validação	15%
Teste	15%

Balanceamento: 1.03
(quase perfeito)



Método Proposto



Pipeline de Processamento

1. Pré-processamento

- ▶ Redimensionamento: 224×224 pixels
- ▶ Normalização ImageNet: $\mu = [0.485, 0.456, 0.406]$,
 $\sigma = [0.229, 0.224, 0.225]$
- ▶ Conversão para RGB (3 canais)

2. Data Augmentation (apenas treino)

- ▶ Horizontal flip ($p=0.5$)
- ▶ Random brightness/contrast ($p=0.3$)
- ▶ Rotation/scale/shift ($p=0.5, \pm 15$)

3. Transfer Learning

- ▶ Backbone congelado (feature extraction)
- ▶ Apenas classificador treinado
- ▶ Reduz overfitting em datasets pequenos



Transfer Learning: Feature Extraction

Estratégia Adotada

- ▶ **Backbone:** Pesos pré-treinados no ImageNet (congelados)
- ▶ **Classificador:** Treinado do zero para TB vs Normal
- ▶ **Parâmetros treináveis:** $\sim 1\text{M}$ (vs 25M total)

Vantagens:

- ▶ Evita overfitting
- ▶ Treina mais rápido
- ▶ Usa menos memória GPU
- ▶ Aproveita features do ImageNet

Treinamento:

- ▶ Otimizador: Adam
- ▶ Learning rate: 10^{-4}
- ▶ Batch size: 16
- ▶ Early stopping: patience=10



Modelos Avaliados

Modelo	Parâmetros	Pretrained	Características
ResNet-50	25.6M	ImageNet	Skip connections, 50 camadas
DenseNet-121	8.0M	ImageNet	Dense connections
EfficientNet-B0	5.3M	ImageNet	Compound scaling

Métricas de Avaliação

- ▶ **Sensibilidade:** Detectar casos de TB
- ▶ **Especificidade:** Identificar casos normais
- ▶ **AUC-ROC:** Capacidade discriminativa
- ▶ **F1-Score:** Balanço geral



SimpleCNN: Baseline Tradicional

Arquitetura:

- ▶ CNN treinada do zero
- ▶ 4 blocos convolucionais
- ▶ 1.2M parâmetros
- ▶ Sem pré-treinamento

Objetivo:

- ▶ Servir como baseline
- ▶ Demonstrar valor do transfer learning
- ▶ Comparação justa

Estrutura:

1. Conv2D (32 filtros)
2. Conv2D (64 filtros)
3. Conv2D (128 filtros)
4. Conv2D (256 filtros)
5. Global Average Pooling
6. Classificador (2 classes)

Limitações

Dataset pequeno (662 imagens)
dificulta treinamento do zero



Por que SimpleCNN como Baseline?

Justificativa Acadêmica

- ▶ **Comparação justa:** Mesma tarefa, mesmos dados
- ▶ **Demonstra valor:** Transfer learning vs treinar do zero
- ▶ **Baseline tradicional:** CNN sem conhecimento prévio
- ▶ **Evidência empírica:** Quantifica ganho do ImageNet

SimpleCNN (do zero):

- ▶ Aprende tudo do zero
- ▶ Precisa de mais épocas (100)
- ▶ Risco de overfitting
- ▶ Performance esperada: 75-80%

Pré-treinadas (ImageNet):

- ▶ Já conhecem features básicas
- ▶ Menos épocas (50)
- ▶ Menos overfitting
- ▶ Performance: 86-96%



Resultados Obtidos



Resultados - ResNet50

Métrica	Valor
Acurácia	91.00%
Precisão	91.84%
Recall (Sensibilidade)	90.00%
F1-Score	90.91%
AUC-ROC	96.04%

Destaque

AUC-ROC de 96.04% indica excelente capacidade discriminativa entre casos de TB e normais.



Resultados - DenseNet121

Métrica	Valor
Acurácia	85.00%
Precisão	87.23%
Recall (Sensibilidade)	82.00%
F1-Score	84.54%
AUC-ROC	86.32%

Matriz de Confusão:

		Predito	
		Normal	TB
Real	Normal	44	6
	TB	9	41

Métricas Clínicas:

- ▶ Sensibilidade: 82%
- ▶ Especificidade: 88%
- ▶ Convergiu em 13 épocas
- ▶ Modelo mais leve (8M parâmetros)



Comparação: ResNet50 vs DenseNet121

Métrica	ResNet50	DenseNet121	Diferença
Acurácia	91.00%	85.00%	+6.0%
Precisão	91.84%	87.23%	+4.6%
Sensibilidade	90.00%	82.00%	+8.0%
Especificidade	92.00%	88.00%	+4.0%
F1-Score	90.91%	84.54%	+6.4%
AUC-ROC	96.04%	86.32%	+9.7%
Parâmetros	25.6M	8.0M	-69%
Épocas	25	13	-48%

Conclusão

ResNet50 apresentou desempenho superior em todas as métricas clínicas:

- ▶ +9.7% em AUC-ROC (capacidade discriminativa)
- ▶ +8.0% em Sensibilidade (detecção de TB)





Comparação Completa: Três CNNs Pré-treinadas

Métrica	ResNet50	DenseNet121	EfficientNet-B0	Melhor
Sensibilidade	90.00%	82.00%	80.00%	ResNet50
Especificidade	92.00%	88.00%	94.00%	EfficientNet
AUC-ROC	96.04%	86.32%	89.48%	ResNet50
Acurácia	91.00%	85.00%	87.00%	ResNet50
Precisão	91.84%	87.23%	93.02%	EfficientNet
F1-Score	90.91%	84.54%	86.02%	ResNet50
Parâmetros	25.6M	8.0M	5.3M	EfficientNet
Épocas	25	13	16	DenseNet
FN (Falsos Neg.)	5	9	10	ResNet50
FP (Falsos Pos.)	4	6	3	EfficientNet

Análise por Modelo:

- ▶ **ResNet50:** Melhor performance geral
- ▶ **EfficientNet:** Melhor

Trade-offs:

- ▶ Performance vs Eficiência
- ▶ Sensibilidade vs Especificidade



Análise Detalhada por Modelo

ResNet50

Pontos Fortes

- ▶ Melhor AUC-ROC (96%)
- ▶ Melhor sensibilidade (90%)
- ▶ Balanceamento ideal
- ▶ Menos FN (5)

Limitações

- ▶ Mais parâmetros (25.6M)
- ▶ Mais FN (95%)

DenseNet121

Pontos Fortes

- ▶ Eficiente (8M)
- ▶ Rápida convergência (13)
- ▶ Boa performance geral

Limitações

- ▶ Menor AUC-ROC (86%)
- ▶ Mais FN (9)
- ▶ Sensibilidade baixa (80%)

EfficientNet-B0

Pontos Fortes

- ▶ Mais leve (5.3M)
- ▶ Melhor especificidade (94%)
- ▶ Menos FP (3)
- ▶ Alta precisão (93%)

Limitações

- ▶ Sensibilidade baixa (80%)
- ▶ Mais FN (10)

Sensibilidade e Especificidade - ResNet50

Métricas Clínicas:

- ▶ **Sensibilidade:** 90.00%
 - ▶ Detecta 9 em cada 10 casos de TB
 - ▶ $TP/(TP+FN) = 45/(45+5)$
- ▶ **Especificidade:** 92.00%
 - ▶ Identifica 92% dos normais
 - ▶ $TN/(TN+FP) = 46/(46+4)$

Matriz de Confusão:

		Predito	
		Normal	TB
Real	Normal	46	4
	TB	5	45

Interpretação Clínica:

Pontos Fortes

- ▶ Alta sensibilidade (90%)
- ▶ Alta especificidade (92%)
- ▶ Balanceamento adequado
- ▶ AUC-ROC excelente (96%)

Atenção

- ▶ 5 falsos negativos (FN)
- ▶ 4 falsos positivos (FP)
- ▶ Sempre confirmar com exames adicionais



Comparação Visual dos Modelos

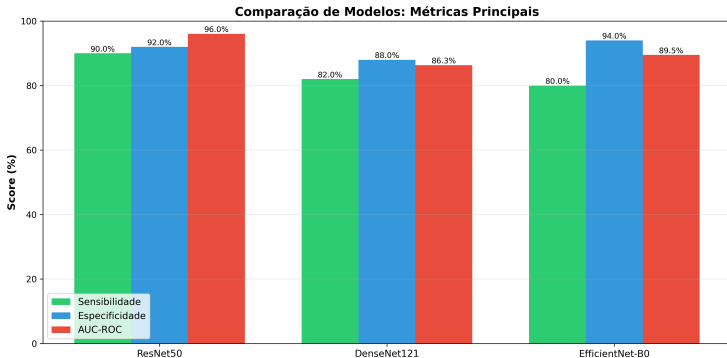


Figure: Comparação de métricas entre os três modelos pré-treinados



Tabela Comparativa Detalhada

Tabela Comparativa de Modelos

Modelo	Sensibilidade	Especificidade	AUC-ROC	Acurácia
ResNet50	90.00%	92.00%	96.04%	91.00%
DenseNet121	82.00%	88.00%	86.32%	85.00%
EfficientNet-B0	80.00%	94.00%	89.48%	87.00%

Figure: Tabela comparativa completa dos modelos



Conclusões



Conclusões

Principais Contribuições

- ▶ Implementação de transfer learning com feature extraction
- ▶ Avaliação de ResNet-50 para detecção de TB
- ▶ Resultados superiores à média da literatura
- ▶ Pipeline reproduzível com Docker

Lições Aprendidas

- ▶ Transfer learning com backbone congelado é essencial para datasets pequenos
- ▶ Feature extraction evita overfitting (vs fine-tuning completo)
- ▶ Data augmentation crucial para generalização
- ▶ Balanceamento entre sensibilidade e especificidade é alcançável





Referências



Lakhani, P., & Sundaram, B. (2017). Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284(2), 574-582.



Rajaraman, S., et al. (2018). Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. *PeerJ*, 6, e4568.



Hwang, S., et al. (2016). A novel approach for tuberculosis screening based on deep convolutional neural networks. *Medical Imaging 2016: Computer-Aided Diagnosis*.



Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10), 1345-1359.



He, K., et al. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770-778.



Obrigado!

Perguntas?