

# Phát hiện bệnh viêm phổi từ ảnh X-quang ngực

\*Note: Sub-titles are not captured in Xplore and should not be used

1<sup>st</sup> Bùi Quang Tuấn  
1771020718  
qtuan280405@gmail.com

2<sup>nd</sup> Hà Tiến Đạt  
1771020137  
datht2545@gmail.com

3<sup>rd</sup> Nguyễn Văn Bảo Ngọc  
1771020510  
zacking112@gmail.com

## 1. GIỚI THIỆU

Viêm phổi là một bệnh nhiễm trùng đường hô hấp dưới do vi khuẩn, virus hoặc nấm gây ra. Bệnh có thể ảnh hưởng nghiêm trọng đến phổi, làm suy giảm chức năng hô hấp và có thể gây tử vong, đặc biệt ở trẻ nhỏ, người cao tuổi và những người có hệ miễn dịch suy yếu. Theo Tổ chức Y tế Thế giới (WHO), viêm phổi là một trong những nguyên nhân hàng đầu gây tử vong trên toàn cầu.

Phát hiện sớm viêm phổi đóng vai trò quan trọng trong việc điều trị kịp thời, giảm nguy cơ biến chứng và cải thiện tiên lượng cho bệnh nhân. Các phương pháp chẩn đoán truyền thống, như thăm khám lâm sàng, xét nghiệm máu và chụp X-quang ngực, thường đòi hỏi bác sĩ có chuyên môn cao để phân tích kết quả. Do đó, việc áp dụng công nghệ tiên tiến để hỗ trợ chẩn đoán nhanh chóng và chính xác hơn là một hướng đi quan trọng.

Trong những năm gần đây, sự phát triển của trí tuệ nhân tạo (AI), đặc biệt là các mô hình học sâu (Deep Learning), đã mở ra cơ hội mới trong lĩnh vực y tế. Một trong những ứng dụng quan trọng là sử dụng mạng nơ-ron tích chập (CNN) để phân tích ảnh X-quang nhằm phát hiện các dấu hiệu của viêm phổi.

Các mô hình phổ biến như DenseNet121 và MobileNetV2 đã được chứng minh có khả năng nhận diện hình ảnh y khoa với độ chính xác cao. Grad-CAM (Gradient-weighted Class Activation Mapping) cũng được sử dụng để trực quan hóa các vùng quan trọng trong ảnh, giúp bác sĩ hiểu rõ hơn về quyết định của mô hình.

Nhờ sự kết hợp giữa học sâu và dữ liệu y khoa, quá trình chẩn đoán viêm phổi có thể được tự động hóa, giúp giảm tải công việc cho bác sĩ và cải thiện chất lượng chăm sóc sức khỏe.

Nghiên cứu này hướng đến việc phát triển một mô hình học sâu hiệu quả để phát hiện viêm phổi từ ảnh X-quang, với các mục tiêu chính sau:

Xây dựng và đánh giá mô hình học sâu có khả năng phân loại ảnh X-quang của bệnh nhân bị viêm phổi và không bị viêm phổi.

Tối ưu hóa mô hình bằng fine-tuning và Grad-CAM để nâng cao độ chính xác và tăng cường khả năng giải thích.

Ứng dụng thực tế: Hỗ trợ bác sĩ trong chẩn đoán, đặc biệt ở các vùng thiếu nhân lực y tế, giúp đưa ra quyết định nhanh hơn và chính xác hơn.

Bằng cách sử dụng AI trong chẩn đoán viêm phổi, nghiên cứu này kỳ vọng sẽ góp phần cải thiện chất lượng dịch vụ y tế và giảm thiểu tỷ lệ tử vong do bệnh này gây ra.

## 2. CÁC NGHIÊN CỨU LIÊN QUAN

### 2.1. Tổng quan các phương pháp truyền thống trong chẩn đoán viêm phổi

Chẩn đoán viêm phổi truyền thống dựa trên sự kết hợp của thăm khám lâm sàng, xét nghiệm sinh học và hình ảnh y khoa. Các phương pháp này đã được sử dụng rộng rãi trong thực hành y khoa, nhưng vẫn tồn tại một số hạn chế như phụ thuộc vào chuyên môn của bác sĩ và thời gian xử lý lâu.

#### • Chẩn đoán lâm sàng

Bác sĩ đánh giá các triệu chứng của bệnh nhân, bao gồm: sốt cao, ớn lạnh, đau ngực, khó thở, mệt mỏi, tím tái ở môi và đầu ngón tay.

Ngoài ra, bác sĩ có thể sử dụng ống nghe để phát hiện các dấu hiệu bất thường ở phổi, như ran ẩm hoặc tiếng thở rít. Tuy nhiên, triệu chứng của viêm phổi có thể tương đồng với các bệnh hô hấp khác (như lao hoặc viêm phế quản), làm cho việc chẩn đoán chỉ dựa vào lâm sàng trở nên khó khăn.

#### • Xét nghiệm máu

Xét nghiệm công thức máu giúp xác định nguyên nhân gây viêm phổi (vi khuẩn hay virus) thông qua:

- Bạch cầu tăng cao: Gợi ý nhiễm khuẩn
- CRP (C-reactive protein) tăng: Đánh giá mức độ viêm
- Cấy máu: Tìm kiếm vi khuẩn gây bệnh và xác định kháng sinh phù hợp

Mặc dù xét nghiệm máu có giá trị trong việc hỗ trợ chẩn đoán, nhưng chúng không thể khẳng định chắc chắn viêm phổi mà chỉ giúp phân biệt giữa các loại nhiễm trùng khác nhau.

#### • Chụp X-quang ngực

Chụp X-quang là phương pháp phổ biến để xác định viêm phổi, cho phép bác sĩ quan sát:

- Vùng mờ bất thường trên phổi (đám mờ phế nang, đông đặc)
- Tràn dịch màng phổi
- Phân biệt viêm phổi với các bệnh phổi khác (lao phổi, phù phổi)

Mặc dù X-quang ngực giúp cung cấp hình ảnh rõ ràng về tổn thương phổi, nhưng việc phân tích ảnh X-quang phụ thuộc vào kinh nghiệm của bác sĩ X-quang, có thể dẫn đến sai sót hoặc chẩn đoán chậm.

- Chụp cắt lớp vi tính (CT scan)

CT scan cung cấp hình ảnh chi tiết hơn so với X-quang, giúp xác định chính xác vị trí và mức độ tổn thương phổi. Phương pháp này thường được sử dụng khi:

- X-quang không đủ rõ ràng
- Bệnh nhân có các bệnh lý phổi khác kèm theo
- Cần phân biệt viêm phổi với khối u phổi

Tuy nhiên, CT scan có chi phí cao hơn và không được áp dụng rộng rãi trong chẩn đoán viêm phổi thông thường.

- Xét nghiệm vi sinh

Các xét nghiệm vi sinh giúp xác định chính xác tác nhân gây viêm phổi, bao gồm:

- Xét nghiệm đờm: Phân tích vi khuẩn hoặc virus
- PCR (Polymerase Chain Reaction): Xác định ADN của tác nhân gây bệnh
- Test kháng nguyên nước tiểu: Phát hiện vi khuẩn như *Streptococcus pneumoniae* hoặc *Legionella pneumophila*

Mặc dù có độ chính xác cao, nhưng xét nghiệm vi sinh thường mất thời gian dài để có kết quả và không phải lúc nào cũng tìm được tác nhân gây bệnh.

## 2.2. Ứng dụng của AI trong phân loại hình ảnh y tế

- Công nghệ hỗ trợ CDHA y tế và vai trò của AI

Trong y khoa, các hệ thống hỗ trợ phát hiện (computer-aided detection – CADe) và hỗ trợ chẩn đoán (computer-aided diagnosis – CADx) được thiết kế để giúp bác sĩ đưa ra quyết định nhanh và chính xác hơn. Cụ thể, các hệ thống này cho phép phân tích và đánh giá các bất thường từ dữ liệu y khoa trong thời gian ngắn. Chúng có thể giúp cải thiện chất lượng hình ảnh y khoa, làm nổi bật các cấu trúc bất thường bên trong cơ thể và thực hiện đo đạc các chỉ số lâm sàng. Các hệ thống CADe/x được xây dựng dựa trên các công nghệ lõi gồm xử lý hình ảnh, thị giác máy tính, và đặc biệt là AI.

Trong 5 năm trở lại đây, sự bùng nổ của dữ liệu lớn và năng lực tính toán đã giúp cho các mô hình AI đạt được những bước tiến vượt bậc. Các đột phá trong nghiên cứu cơ bản và nghiên cứu ứng dụng về AI trong y tế liên tiếp được công bố và đưa vào triển khai thực tế. Nói một cách vắn tắt, các mô hình AI có thể tham gia hỗ trợ bác sĩ trong toàn bộ quy trình khám chữa bệnh lâm sàng dựa trên hình ảnh y tế. AI cho phép tạo ra các hình ảnh y khoa nhanh hơn, chất lượng hơn với giá thành rẻ hơn. Việc phân tích, chẩn đoán bệnh và tự động xuất báo cáo... cũng đều có thể đảm nhiệm bởi các thuật toán AI.

Năng lực của một hệ thống AI trong CDHA y tế có thể chia làm 3 mức độ:

- (1) Detection: phát hiện các bất thường tiềm tàng từ hình ảnh, phân loại giữa các hình ảnh bệnh nhân mắc bệnh và bệnh nhân không mắc bệnh. Chức năng này của AI giúp giải quyết các bài toán sàng lọc diện rộng

hoặc tối ưu việc phân luồng bệnh nhân theo tính ưu tiên.

- (2) Characterization: đặc tả các tính chất của tổn thương, tiến hành khoanh vùng, phân loại mức độ nguy hiểm theo các tiêu chuẩn y khoa. Ví dụ phân loại mức độ ung thư hoá của một tổn thương.
- (3) Monitoring: giám sát và đánh giá sự tiến triển của các tổn thương trên hình ảnh. Theo dõi và đánh giá theo thời gian thực các tình huống khẩn cấp nguy hiểm tới tính mạng của người bệnh.

- Những thành tựu bước đầu tại Việt Nam

Nghiên cứu và ứng dụng AI trong CDHA y tế tại Việt Nam đang ở giai đoạn đầu. Một số ứng dụng AI hỗ trợ chẩn đoán do các công ty công nghệ quốc tế phát triển đã được triển khai tại một số bệnh viện. Tuy nhiên, dữ liệu người Việt có tính đặc thù. Điều này làm giảm hiệu quả của các thuật toán AI vốn được huấn luyện dựa trên dữ liệu. Một số tập toàn trong nước như Vingroup, Viettel hay FPT cũng đã từng bước phát triển các giải pháp thông minh hỗ trợ CDHA y tế.

Mới đây, hệ thống trí tuệ nhân tạo VinDr do Trung tâm Xử lý ảnh y tế, trực thuộc VinBigdata phát triển cho phép CDHA về bệnh lý phổi trên ảnh X-quang lồng ngực, chẩn đoán ung thư vú trên ảnh X-quang tuyến vú, xác định các bất thường trên hình ảnh CT/MRI sọ não và phát hiện các bất thường trên phim chụp X-quang cột sống đã được thử nghiệm lâm sàng tại một số bệnh viện lớn như Bệnh viện Trung ương Quân đội 108, Bệnh viện Đại học Y Hà Nội, Bệnh viện Đa khoa Quốc tế Vinmec.



Fig. 1. VinDr hỗ trợ chẩn đoán ảnh X-quang cột sống với độ chính xác cao

Thành công ban đầu của VinDr đã khẳng định trình độ của các nhà khoa học Việt Nam. Số lượng các hệ thống AI được phát triển bài bản, áp dụng rộng rãi và tạo được ảnh hưởng tích cực lên hiệu quả chẩn đoán bệnh hiện còn hạn chế. Chính vì vậy, rất cần phát triển những công cụ AI mới, với sự tham gia, hợp tác của nhiều đơn vị nghiên cứu và phát triển công nghệ trong nước.

## 2.3. Một số mô hình đã được sử dụng trước đây cho bài toán này

ResNet (ResNet50, ResNet101, ResNet152): Sử dụng residual connections giúp tránh vấn đề vanishing gradient.

DenseNet (DenseNet121, DenseNet169, DenseNet201): Kết nối dày đặc giữa các lớp giúp tăng khả năng truyền thông tin.

MobileNetV2: Nhẹ, tối ưu cho thiết bị di động, sử dụng depthwise separable convolution.

Vision Transformers (ViT): Sử dụng self-attention thay vì convolution để học các đặc trưng.

Tiêu chí	DenseNet121	MobileNetV2	ResNet50	ViT (Vision Transformer)
Kích thước mô hình	~8MB	~3.4MB	~25MB	~85MB
Tốc độ suy luận	Trung bình	Nhanh (tối ưu cho thiết bị di động)	Chậm hơn do số tham số lớn	Chậm (yêu cầu phần cứng mạnh)
Độ chính xác	Cao (~85-95% trên các dataset y tế)	Trung bình (~80-90%)	Cao (~85-95%)	Cao nếu có dữ liệu lớn (>100K ảnh)
Khả năng học đặc trưng	Rất tốt (các layer kết nối chặt chẽ)	Khá tốt nhưng có thể mất thông tin	Tốt nhờ residual learning	Rất tốt nhưng cần dữ liệu lớn
Overfitting	Ít bị do sharing feature maps	Ít bị, thích hợp cho tập dữ liệu nhỏ	Có thể bị nếu dữ liệu ít	Dễ bị nếu dữ liệu không đủ lớn
Ứng dụng trong y tế	Rất phổ biến, hiệu quả cao	Tốt trên thiết bị di động	Tốt nhưng nặng hơn	Chưa phổ biến, cần nghiên cứu thêm

Fig. 2. So sánh một vài đặc trưng giữa các mô hình

### 3. DỮ LIỆU VÀ PHƯƠNG PHÁP NGHIÊN CỨU

#### 3.1. Dữ liệu

Sử dụng tập dữ liệu Chest X-ray Images (Pneumonia) được lấy Kaggle.com qua đường link ''

Số lượng ảnh lớn hơn 5000, được chia thành 3 phần cho train, val và test, trong đó:

- 85% dành cho train
- 5% dành cho val
- 10% dành cho test

Các ảnh đều được dán nhãn để phân biệt giữa các trường hợp bình thường và mắc bệnh viêm phổi, để phục vụ cho việc kiểm tra về sự chính xác của mô hình

#### 3.2. Phương pháp tiếp cận

##### a) Mô hình DenseNet121

Cấu trúc của DenseNet dựa trên một ý tưởng đơn giản nhưng mạnh mẽ: tăng cường kết nối giữa các lớp trong mạng nơ-ron. Trong các kiến trúc mạng nơ-ron truyền thống, mỗi lớp chỉ nhận đầu vào từ lớp liền trước và gửi đầu ra đến lớp liền sau. Tuy nhiên, trong DenseNet, mỗi lớp nhận đầu vào từ tất cả các lớp trước đó và gửi đầu ra đến tất cả các lớp tiếp theo. Điều này tạo ra một mạng lưới dày đặc, nơi mà các đặc trưng được chia sẻ và tái sử dụng liên tục giữa các lớp.

$$x \rightarrow [x, f_1(x), f_2(x, f_1(x)), f_3(x, f_1(x), f_2(x, f_1(x))), \dots]$$

##### (\*) Dense Block

Mỗi Dense Block gồm nhiều lớp kết nối dày đặc với nhau. Đặc trưng của mỗi lớp trong một Dense

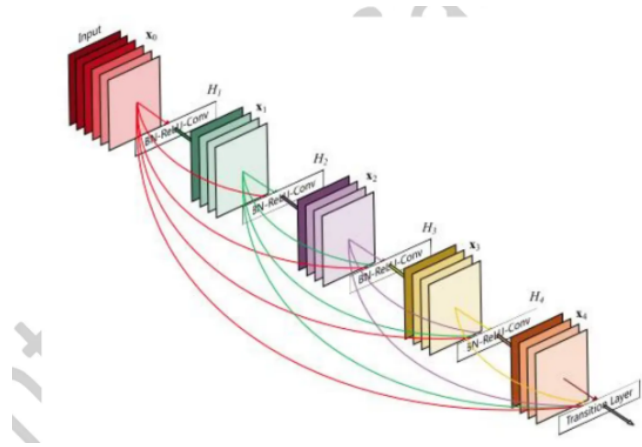


Fig. 3. Mô phỏng cấu trúc của DenseNet

Block là việc nhận đầu vào từ tất cả các lớp trước đó, sau đó xuất đầu ra để sử dụng cho các lớp sau. Mỗi lớp trong một Dense Block thường bao gồm một lớp chuẩn hóa theo batch (Batch Normalization), một lớp kích hoạt ReLU, và một lớp tích chập (Convolutional Layer).

Các Dense Block giúp giảm độ phức tạp của mô hình mà vẫn giữ được độ chính xác cao nhờ vào việc tái sử dụng các đặc trưng đã học.

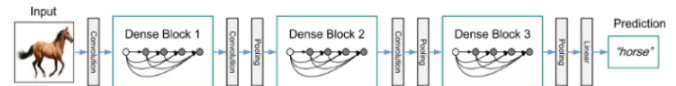


Fig. 4. Ví dụ về cách Dense Block hoạt động

##### (\*) Transition Layer

Giữa các Dense Block, có các lớp chuyển tiếp giúp giảm kích thước của bản đồ đặc trưng (feature map) bằng cách sử dụng các lớp tích chập 1x1 và phép gộp trung bình (Average Pooling). Transition Layer đóng vai trò quan trọng trong việc kiểm soát kích thước và độ phức tạp của mô hình, ngăn chặn sự phình to của số lượng đặc trưng khi mạng trở nên quá sâu.

##### b) Fine-tuning mô hình và tối ưu hóa bằng EarlyStopping.

##### (\*) Fine-tuning mô hình

- Chọn mô hình pre-trained  
Sử dụng DenseNet121 đã được huấn luyện trên tập ImageNet.
- Đóng băng một số lớp đầu của mô hình
- Thêm các lớp mới cho bài toán phát hiện viêm phổi  
GlobalAveragePooling2D(): Giúp giảm chiều dữ liệu đầu ra của CNN trước khi đưa vào lớp Dense.  
Dense(128, activation='relu'): Một lớp Fully Connected với 128 neuron và activation function ReLU.

Dropout(0.5): Giúp giảm overfitting bằng cách ngẫu nhiên tắt 50% số neuron.

Dense(1, activation='sigmoid'): Đầu ra của mô hình với xác suất dự đoán bệnh viêm phổi.

- Compile mô hình

Sử dụng Adam optimizer với learningrate=0.0001 để giúp hội tụ nhanh hơn.

binary\_crossentropy: Vì đây là bài toán phân loại nhị phân (có viêm phổi hoặc không).

(\*) Tối ưu hóa bằng EarlyStopping

EarlyStopping giúp dừng quá trình huấn luyện sớm nếu mô hình không cải thiện sau một số epoch nhất định.

Tránh overfitting bằng cách lưu lại trọng số tốt nhất của mô hình.

Tham số chính:

- monitor='valloss': Theo dõi giá trị valloss để quyết định khi nào dừng.

- patience=5: Nếu valloss không giảm trong 5 epoch liên tiếp, huấn luyện sẽ dừng.

- restore\_best\_weights=True: Giúp phục hồi trọng số tốt nhất trước khi dừng.

c) Trực quan hóa vùng quan trọng với Grad-CAM

(\*) Tạo một mô hình phụ để lấy output của một lớp CNN cụ thể (conv5-block16-concat trong DenseNet121) và đầu ra của mô hình gốc.

(\*) Tính gradient của đầu ra mô hình với các feature map

(\*) Cải tiến bằng ReLU vào gradient

Gradient có thể có giá trị âm và dương, nhưng chỉ giá trị dương mới quan trọng trong Grad-CAM. Bằng cách sử dụng tf.maximum(grads, 0), ta chỉ giữ lại gradient dương.

(\*) Tính trung bình gradient theo chiều không gian

Điều này giúp tính trọng số quan trọng của mỗi feature map.

(\*) Tạo heatmap

Nhân trọng số pooled-grads với các feature map.

Chuẩn hóa giá trị về khoảng [0,1].

(\*) Chồng heatmap lên ảnh gốc

Resize heatmap về kích thước ảnh gốc.

Áp dụng cv2.COLORMAP-JET để tạo màu sắc cho heatmap.

Dùng cv2.addWeighted() để kết hợp với ảnh gốc.

#### 4. THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

##### 4.1. Cấu hình thực nghiệm

a) Thông số huấn luyện

(\*) Số epoch: epochs=20

- Mô hình sẽ huấn luyện tối đa 20 epoch.

- Sử dụng EarlyStopping với patience=5, nên nếu mô hình không cải thiện sau 5 epoch, quá trình huấn luyện sẽ dừng sớm.

(\*) Batch size: batch-size=32

- Mỗi lần cập nhật trọng số, mô hình sẽ xử lý 32 ảnh.

- Batch size vừa phải giúp cân bằng giữa tốc độ huấn luyện và hiệu suất mô hình.

(\*) Learning rate: learning-rate=0.0001

- Sử dụng tốc độ học 0.0001 (1e-4) khi huấn luyện.

- Kết hợp ReduceLROnPlateau, giảm learning rate theo factor=0.5 khi val-loss không cải thiện sau 3 epoch (tối thiểu 1e-7).

(\*) Optimizer: Adam

- Sử dụng Adam optimizer (Adam(learning-rate=0.0001)).

- Adam kết hợp giữa momentum và RMSprop, giúp tối ưu hóa nhanh chóng và ổn định.

b) Phần cứng và môi trường

(\*) Phần cứng:

- GPU: NVIDIA Tesla T4

- Bộ nhớ RAM: 16GB

- Nền tảng: Google Colab

(\*) Môi trường phần mềm:

- Hệ điều hành: Ubuntu 20.04 (Google Colab)

- Phiên bản Python: 3.9+

- Phiên bản TensorFlow: 2.x

(\*) Thư viện hỗ trợ:

- TensorFlow/Keras: Xây dựng và huấn luyện mô hình học sâu

- OpenCV (cv2): Xử lý và tiền xử lý ảnh

- Matplotlib & Seaborn: Vẽ biểu đồ, trực quan hóa Grad-CAM

- scikit-learn: Đánh giá mô hình (confusion matrix, classification report)

- Google Drive (drive): Lưu trữ và tải dữ liệu từ Google Drive

##### 4.2. Kết quả đánh giá

a) Hiệu suất mô hình

(\*) Precision:

- Công thức:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

- Normal: 0.94 → 94% mẫu dự đoán là "Normal" thực sự đúng.

- Pneumonia: 0.93 → 93% mẫu dự đoán là "Pneumonia" thực sự đúng.

(\*) Recall:

- Tỷ lệ phát hiện đúng bệnh trong tổng số bệnh nhân thực sự có bệnh.

- Công thức:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

- Normal: 0.88 → Chỉ 88% trường hợp "Normal" được nhận diện đúng, một số bị dự đoán nhầm là "Pneumonia".

- Pneumonia: 0.96  $\rightarrow$  96% ca "Pneumonia" được nhận diện đúng.

(\*) F1-score:

- Trung bình điều hòa của Precision và Recall, giúp cân bằng giữa hai chỉ số này.
- Công thức:

$$F1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

- Normal: 0.91
- Pneumonia: 0.95

(\*) Support:

Số lượng mẫu thực tế trong mỗi lớp:

- Normal: 234 ảnh.
- Pneumonia: 390 ảnh.

b) Ma trận nhầm lẫn và classification report

Mô hình hoạt động với số epoch = 20 đã cho ra những kết quả tích cực được thể hiện ở ma trận nhầm lẫn và bảng hiệu suất.

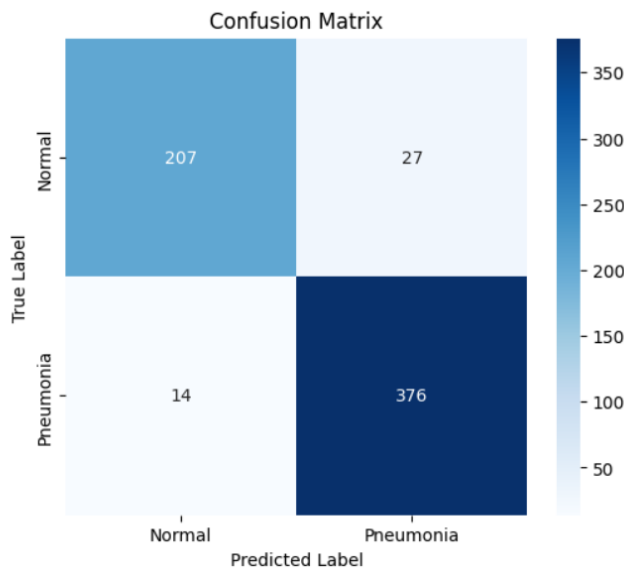


Fig. 5. Confusion Matrix

- Số trường hợp Normal dự đoán đúng là:  $207 / 234 = 88,4\%$
- Số trường hợp Pneumonia dự đoán đúng là:  $376 / 390 = 96,4\%$

	precision	recall	f1-score	support
Normal	0.94	0.88	0.91	234
Pneumonia	0.93	0.96	0.95	390
accuracy			0.93	624
macro avg	0.93	0.92	0.93	624
weighted avg	0.93	0.93	0.93	624

Fig. 6. Classification Report

## 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1. Kết quả đạt được

Mô hình chạy tốt với các tập dữ liệu y tế, không có ràng buộc hay yêu cầu về kích cỡ của dữ liệu.

Độ chính xác mô hình đạt được trên cả tập huấn luyện và test đều trên 90%

Xác định vùng có tỷ lệ cao bị thương tổn trên phổi và đưa ra dự đoán về tỷ lệ mắc bệnh viêm phổi

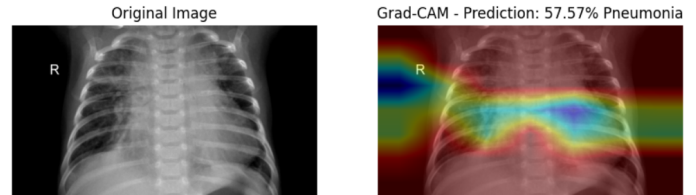


Fig. 7. Một ví dụ về kết quả chạy của mô hình

### 5.2. Hạn chế của nghiên cứu

Tốc độ xử lý của mô hình còn chậm, sẽ mất nhiều thời gian để xử lý những tập dữ liệu lớn.

Mô hình chỉ mới đưa ra tỷ lệ mắc bệnh và chỉ ra vùng có khả năng thương tổn. Vẫn chưa chỉ ra được rõ từng vị trí có vấn đề trên phổi.

Mô hình chỉ mới ở mức độ tìm hiểu và thử nghiệm chưa áp dụng được trong lĩnh vực y tế.

### 5.3. Hướng phát triển

a) Cải thiện hiệu suất mô hình

(\*) Thử nghiệm các mô hình mạnh hơn

- Sử dụng mô hình mới hơn: Thử nghiệm với các mô hình CNN tiên tiến hơn như EfficientNet, Vision Transformer (ViT) hoặc ConvNeXt.
- Ensemble Learning: Kết hợp nhiều mô hình để tăng độ chính xác.

(\*) Điều chỉnh hyperparameter

- Tăng số epoch
- Thử nghiệm các batch size khác nhau (16, 64 thay vì 32).
- Tối ưu learning rate bằng Learning Rate Scheduler hoặc Cyclical Learning Rate (CLR).

(\*) Fine-tuning hiệu quả hơn

- Fine-tune nhiều lớp hơn để cải thiện khả năng học.
- Sử dụng Layer-wise Learning Rate Adaptation để tinh chỉnh từng phần của mô hình một cách tối ưu.

b) Cải thiện khả năng giải thích và trực quan hóa

(\*) Tăng cường Grad-CAM

- Sử dụng Grad-CAM++ hoặc Score-CAM để có bản đồ nhiệt chi tiết hơn.
- Kết hợp Grad-CAM với LIME hoặc SHAP để hiểu rõ hơn cách mô hình ra quyết định.

(\*) Phát hiện vùng bất thường trên ảnh

- Kết hợp với segmentation models (như U-Net, Mask R-CNN) để xác định chính xác vị trí viêm phổi trên ảnh X-quang.
- c) Mở rộng dữ liệu và khả năng tổng quát hóa
  - (\*) Tăng cường dữ liệu
    - Có thể thử CutMix, MixUp, hoặc GANs (Generative Adversarial Networks) để tạo thêm ảnh X-quang nhân tạo.
  - (\*) Sử dụng dữ liệu đa nguồn
    - Kết hợp nhiều bộ dữ liệu X-quang khác nhau.
    - Thử nghiệm mô hình trên dữ liệu đa trung tâm để tăng tính tổng quát.
  - (\*) Phát triển mô hình đa nhiệm
    - Có thể mở rộng để phát hiện loại viêm phổi (do virus hay vi khuẩn) hoặc phát hiện thêm các bệnh khác (lao phổi, xơ phổi, ung thư phổi, COVID-19,...).

### **LỜI CẢM ƠN**

Xin chân thành gửi lời cảm ơn tới thầy Lê Trung Hiếu đã đóng góp ý kiến trong quá trình nghiên cứu. Và gửi lời cảm ơn đến các bạn đã góp sức trong quá trình hoàn thành nghiên cứu và cho ra bài báo cáo này

### **TÀI LIỆU THAM KHẢO**

- (1) Võ Đức Quang, Nguyễn Hải Yến, Võ Hồng Mận: *Ứng dụng kỹ thuật học sâu trong hỗ trợ chẩn đoán bệnh viêm phổi thông qua ảnh chụp X-quang*
- (2) Ngô Hoàng Ân, Lê Minh Thanh, Lê Trạch Đình: *THIẾT KẾ CHUỖNG TRÌNH CHẨN ĐOÁN DỰA TRÊN X-QUANG NGỰC*
- (3) VinDr AI hỗ trợ chẩn đoán sớm nhiều bệnh lý nguy hiểm của người Việt