

BỘ NÔNG NGHIỆP VÀ MÔI TRƯỜNG
PHÂN HIỆU TRƯỜNG ĐẠI HỌC THỦY LỢI
BỘ MÔN CÔNG NGHỆ THÔNG TIN



BÁO CÁO KHAI PHÁ DỮ LIỆU

ĐỀ TÀI : Khai phá dữ liệu ảnh X-quang phổi (Chest -Xray) để phát hiện viêm phổi bằng Deep_Learning

Giảng viên hướng dẫn:	ThS: Vũ Thị Hạnh
Lớp: S25-64CNTT	Thành viên:
	Nguyễn Hoàng Tân - 2251068249
	Võ Quốc Pháp - 2251068226 Nguyễn Hoàng Anh - 2251068173

Ngày 31 Tháng 10 Năm 2025

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN VÀ GIỚI THIỆU ĐỀ TÀI.....	6
1.1. Lý do chọn đề tài.....	6
1.2. Mục tiêu nghiên cứu	6
1.3. Bộ dữ liệu nghiên cứu	7
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	8
2.1. Cơ sở lý thuyết về Mạng Nơ-ron Tích chập (CNN).....	8
2.2. Cơ sở lý thuyết về mô hình ResNet50	9
2.3. Cơ sở lý thuyết về mô hình Xception	13
Chi tiết từng khối trong hình:	17
CHƯƠNG 3: TIỀN XỬ LÝ DỮ LIỆU	18
3.1. Giới thiệu	18
3.2. Các bước tiền xử lý cơ bản.....	19
3.3. Tiền xử lý cho các mô hình	19
3.3.1. Tiền xử lý trong mô hình CNN cơ bản	19
3.3.2. Tiền xử lý trong các mô hình CNN hiện đại (ResNet50, Xception, EfficientNet)	20
CHƯƠNG 4 PHƯƠNG PHÁP KHAI PHÁ DỮ LIỆU VÀ.....	22
MÔ HÌNH MACHINE LEARNING.....	22
4.1. Tổng quan.....	22
4.2. Khai phá dữ liệu.....	23
4.2.1 Phân tích phân bố nhãn (Label Distribution):	23
4.2.2. Phân tích đặc điểm ảnh (Image Properties):.....	24
4.2.3 Phát hiện và loại bỏ dữ liệu lỗi:.....	26
4.2.4. Trực quan hóa dữ liệu:	27
4.2.5. Phát hiện dữ liệu trùng lặp hoặc tương tự (Duplicate/Similar Images): ..	28
CHƯƠNG 5: KẾT QUẢ VÀ ĐÁNH GIÁ MÔ HÌNH.....	29
5.1. Tiêu chí và phương pháp đánh giá.....	29
5.2. Kết quả mô hình CNN cơ bản của tác giả	29
5.2.1. Kết quả huấn luyện mô hình CNN normal.....	29
5.2.2. Kết quả đánh giá trên tập kiểm thử của CNN normal.....	31
5.2.3. Báo cáo phân loại	32
5.3.1. Cấu trúc mô hình	32

5.3.2. Kết quả huấn luyện của ResNet50	33
5.3.3. Kết quả đánh giá trên tập kiểm thử của ResNet50	33
5.3.4. Báo Cáo Phân Loại.....	34
5.3.5. Ma trận nhầm lẫn của Resnet50	35
5.4. Mô hình Xception	36
5.4.1. Cấu trúc mô hình	36
5.4.2. Kết quả huấn luyện của Xception	37
5.4.3. Báo cáo phân loại	38
5.4.4. Ma trận nhầm lẫn (Confusion Matrix)	39
5.4.5. Nhận xét tổng quan.....	40
5.5. Kết quả mô hình EfficientNetB0	40
5.5.1. Cấu trúc mô hình	40
5.5.2. Kết quả huấn luyện mô hình EfficientNetB0	41
5.5.3. Kết quả đánh giá trên tập kiểm thử	43
5.5.4. Báo cáo phân loại	44
5.5.5. Ma trận nhầm lẫn (Confusion Matrix)	44
5.6. So sánh và kết luận tổng hợp các mô hình	45
5.7. Triển khai dự án với model EfficientNetB0	46
5.7.1. Kiến trúc tổng thể	46
5.7.2 Vai trò Backend (FastAPI)	46
5.7.3 Vai trò Frontend (Vite + React)	47
5.7.4 Triển khai & vận hành.....	47
5.7.5 Lý do chọn EfficientNetB0 để triển khai	47
5.7.6. Giao diện hệ thống và minh họa kết quả triển khai.....	47
CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	50
6.1. Kết luận.....	50
6.2. Hạn chế	50
6.3. Hướng phát triển	50
TÀI LIỆU THAM KHẢO	51

MỤC LỤC ẢNH

<i>Hình 1 bộ dữ liệu xray</i>	<i>7</i>
<i>Hình 2 Giới thiệu về mô hình CNN.....</i>	<i>8</i>
<i>Hình 3 Giới thiệu về mô hình ResNet 50.....</i>	<i>9</i>
<i>Hình 4 vanish gradient và exploding gradient.....</i>	<i>10</i>
<i>Hình 5 kiến trúc của mô hình ResNet50</i>	<i>10</i>
<i>.....</i>	<i>12</i>
<i>Hình 6: Kiến trúc của mô hình ResNet50.2</i>	<i>12</i>
<i>Hình 7 Giới thiệu mô hình Xception</i>	<i>13</i>
<i>Hình 8 lớp tích chập tách biệt theo chiều sâu.....</i>	<i>14</i>
<i>Hình 9 sự khác biệt giữa phép tích chập thông thường và tích chập tách biệt theo chiều sâu</i>	<i>15</i>
<i>Hình 10 Giới thiệu về mô hình efficientNet</i>	<i>16</i>
<i>Hình 11 Kiến trúc mô hình efficientNet.....</i>	<i>17</i>
<i>Hình 12: Quy trình tiền xử lý dữ liệu CNN.....</i>	<i>18</i>
<i>Hình 13: Minh họa một ảnh X-quang điển hình trong tập dữ liệu.....</i>	<i>25</i>
<i>Hình 14: Minh họa Grad-CAM.....</i>	<i>27</i>
<i>Hình 15: Accuracy của CNN normal</i>	<i>29</i>
<i>Hình 16: Loss của CNN normal.....</i>	<i>30</i>
<i>Hình 17: AUC (ROC) của CNN normal</i>	<i>31</i>
<i>Hình 18: Classification Report của Normal CNN</i>	<i>32</i>
<i>Hình 19: Kiến trúc ResNet50</i>	<i>32</i>
<i>Hình 20: Số lần huấn luyện của Resnet50</i>	<i>33</i>
<i>Hình 21: AUC (ROC) của ResNet50</i>	<i>34</i>
<i>Hình 22: Classification Report của ResNet50.....</i>	<i>34</i>
<i>Hình 23: Ma trận nhầm lẫn (Confusion Matrix) của ResNet50.....</i>	<i>35</i>
<i>Hình 24: Cấu trúc mô hình Xception</i>	<i>36</i>
<i>Hình 25: Biểu đồ Training vs Validation Accuracy.....</i>	<i>37</i>
<i>Hình 26: Biểu đồ Training vs Validation Loss</i>	<i>38</i>
<i>Hình 27: Classification Report chi tiết của mô hình Xception.....</i>	<i>38</i>
<i>28: Ma trận nhầm lẫn ba lớp (Normal – Bacteria – Virus) của Xception.....</i>	<i>39</i>

Hình 29: OVERALL METRICS của Xception.....	40
Hình 30: Cấu trúc EfficientNetB0 backbone.	41
Hình 31: Biểu đồ Training vs Validation Accuracy.....	42
Hình 32: Biểu đồ Training vs Validation Loss	43
Hình 33:OVERALL METRICS của EfficientNetB0	43
Hình 34: Classification Report chi tiết của EfficientNetB0	44
Hình 35; Ma trận nhầm lẫn ba lớp (Normal – Bacteria – Virus) của EfficientNetB0.....	45
Hình 36: Giao diện fontend Upload Ảnh.....	48
Hình 37: Trang Fontend hiển thị kết quả.....	49
Hình 38: Trang Fontend hiển thị lịch sử sử dụng	49

CHƯƠNG 1: TỔNG QUAN VÀ GIỚI THIỆU ĐỀ TÀI

1.1. Lý do chọn đề tài

Trong thời đại hiện nay, trí tuệ nhân tạo (AI) và học sâu (Deep Learning) đang ngày càng đóng vai trò quan trọng trong lĩnh vực y học, đặc biệt là trong chẩn đoán hình ảnh (Medical Imaging). Các phương pháp truyền thống đòi hỏi bác sĩ phải đọc và phân tích từng tấm phim X-quang, vừa tốn thời gian, vừa dễ xảy ra sai sót do yếu tố con người. Trong số các bệnh về đường hô hấp, viêm phổi (Pneumonia) là một căn bệnh phổ biến và nguy hiểm, gây tử vong cao, đặc biệt ở trẻ em và người cao tuổi. Theo Tổ chức Y tế Thế giới (WHO), viêm phổi là nguyên nhân gây ra hơn 2 triệu ca tử vong mỗi năm. Việc phát hiện sớm và chẩn đoán chính xác viêm phổi là yếu tố quyết định để điều trị kịp thời và giảm tỷ lệ tử vong. Tuy nhiên, tại nhiều cơ sở y tế, đặc biệt ở vùng nông thôn, thiếu bác sĩ X-quang và thiết bị phân tích hiện đại khiến việc chẩn đoán trở nên khó khăn. Do đó, nhóm chúng tôi cụ thể là nhóm 8 quyết định thực hiện đề tài: “Khai phá dữ liệu ảnh X-quang phổi (Chest X-ray) để phát hiện viêm phổi bằng học sâu (Deep Learning)” với sứ mệnh và mục tiêu là xây dựng hệ thống trí tuệ nhân tạo có khả năng phân loại tự động ảnh X-quang thành các nhóm sau :

- Phổi bình thường
- Phổi viêm do vi khuẩn
- Phổi viêm do virus

1.2. Mục tiêu nghiên cứu

Đề tài hướng tới việc ứng dụng các mô hình học sâu (Deep Learning) với mô hình CNN (Convolutional Neural Network), Resnet 50, XceptionNetv2, EfficientNet để tự động nhận diện và phân loại bệnh viêm phổi từ hình ảnh X-quang.

Các mục tiêu cụ thể gồm:

1. Tìm hiểu cơ sở lý thuyết về học sâu và các kiến trúc CNN hiện đại (ResNet50, Xception, EfficientNet).
2. Thu thập và tiền xử lý dữ liệu ảnh X-quang phổi từ nguồn công khai (ví dụ: Kaggle, NIH).
3. Huấn luyện và đánh giá mô hình CNN cho bài toán phân loại các lớp: Normal / Pneumonia_bacteria / Pneumonia_virus.
4. So sánh hiệu năng của các mô hình học sâu (CNN, ResNet50, Xception, EfficientNet).

5. Trực quan hóa kết quả bằng Grad-CAM để xác định vùng mô hình tập trung khi dự đoán.
6. Đề xuất hướng cải tiến nhằm nâng cao độ chính xác và ứng dụng thực tế trong chẩn đoán y tế.

1.3. Bộ dữ liệu nghiên cứu



Hình 1 bộ dữ liệu xray

- Bộ dữ liệu sử dụng trong đề tài được công bố công khai trên nền tảng Kaggle – Chest X-Ray Pneumonia Dataset, do Trung tâm Phụ nữ và Trẻ em Quảng Châu (Guangzhou Women and Children's Medical Center) cung cấp.
- Với cấu trúc thư mục được 5.863 ảnh X-quang ngực với định dạng jpeg được chia thành 2 nhóm chính với thư mục con (NORMAL, PENUMONIA):

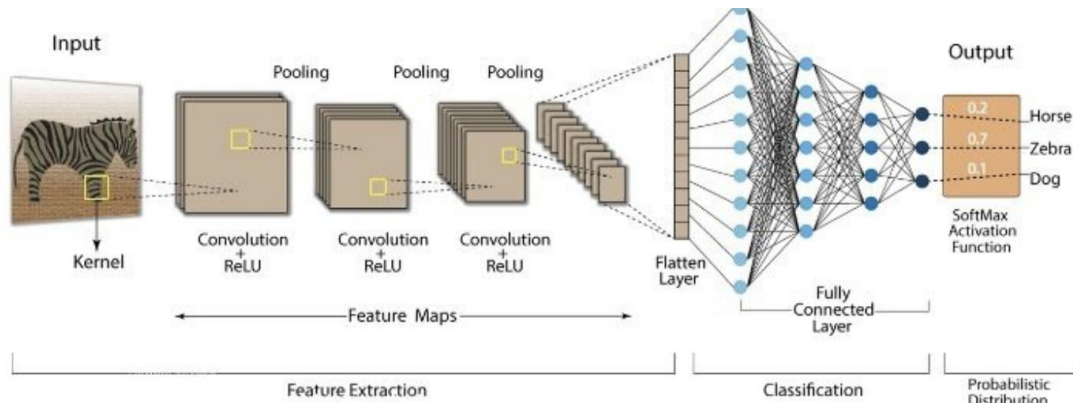
Nhóm dữ liệu	Mô tả	Số lượng ảnh
Train	Dữ liệu huấn luyện mô hình	5.000
Validation	Dữ liệu hiệu chỉnh tham số	800
Test	Dữ liệu kiểm thử mô hình	1.000

- Các ảnh X-quang ngực được chụp ở tư thế trước - sau (PA view) của bệnh nhi từ 1 đến 5 tuổi. Dữ liệu được thu thập từ hồ sơ hồi cứu y tế tại Trung tâm Phụ nữ và Trẻ em Quảng Châu, trong quá trình chăm sóc lâm sàng thông thường.
- Tất cả ảnh X-quang đều được kiểm định chất lượng ban đầu, loại bỏ các ảnh:
 - Mờ, nhiễu, hoặc sai tư thế
 - Có vật thể cản trở (như thiết bị y tế)
- Mỗi ảnh được chẩn đoán độc lập bởi hai bác sĩ chuyên khoa X-quang nhi.

- Bộ dữ liệu kiểm định được xác minh lại bởi bác sĩ thứ ba để đảm bảo độ tin cậy trước khi đưa vào hệ thống AI huấn luyện.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Cơ sở lý thuyết về Mạng Nơ-ron Tích chập (CNN)



Hình 2 Giới thiệu về mô hình CNN

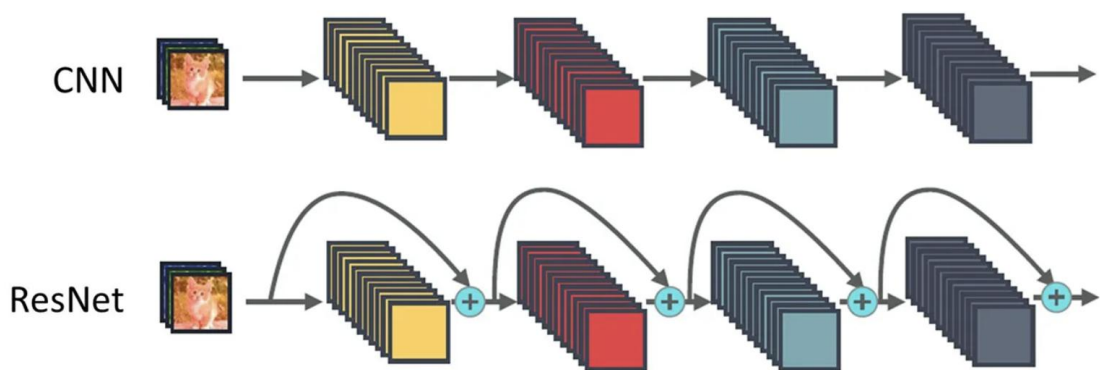
- Khái niệm về CNN (Convolutional Neural Network) là một loại mạng học sâu được thiết kế đặc biệt cho dữ liệu dạng hình ảnh. Khác với mạng truyền thống (MLP), CNN khai thác tính chất không gian của ảnh thông qua các phép tích chập (convolution) để tự động học ra đặc trưng quan trọng.
- Với mô hình CNN có các thành phần chính như sau:

Các lớp của mô hình CNN	Ý nghĩa
Convolutional Layer	Trích xuất đặc trưng bằng các bộ lọc (kernel) kích thước nhỏ (3×3 , 5×5). Mỗi bộ lọc học ra một loại đặc trưng (cạnh, góc, vùng sáng tối,...).
Activation Function (ReLU)	Tạo phi tuyến tính, loại bỏ giá trị âm.
Pooling Layer	Giảm kích thước không gian, giữ đặc trưng quan trọng nhất. Gồm hai loại: <i>Max pooling</i> và <i>Average pooling</i> .

Fully Connected Layer	Tổng hợp các đặc trưng học được để phân loại đầu ra.
Softmax Layer	Tổng hợp các đặc trưng học được để phân loại đầu ra.

- Cùng với đó là ưu điểm của CN bao gồm:
 1. Tự động học đặc trưng mà không cần trích xuất thủ công.
 2. Hoạt động tốt với dữ liệu ảnh lớn, phức tạp.
 3. Dễ mở rộng với các kiến trúc khác nhau như ResNet, Xception, EfficientNet.

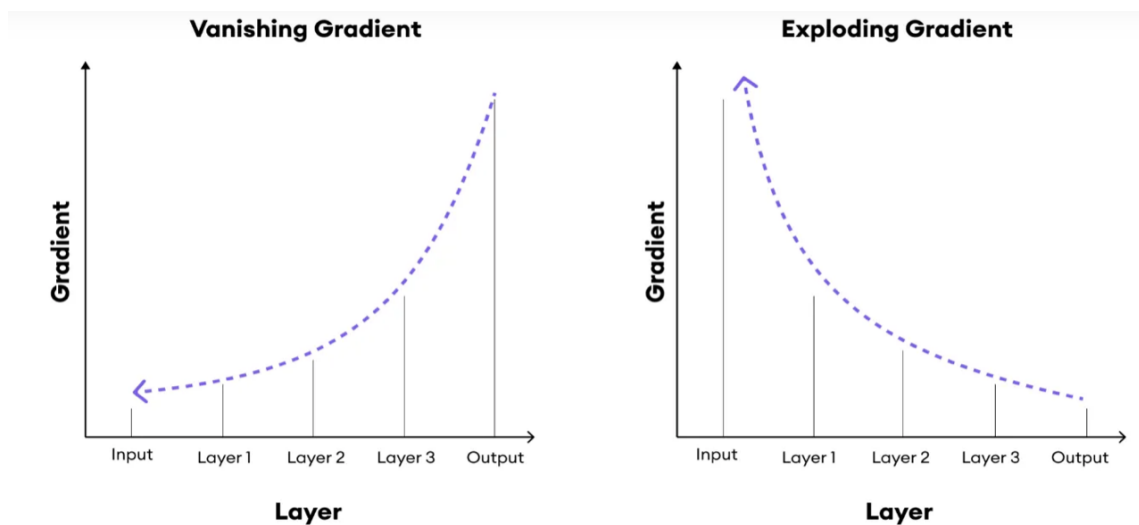
2.2. Cơ sở lý thuyết về mô hình ResNet50



Hình 3 Giới thiệu về mô hình ResNet 50

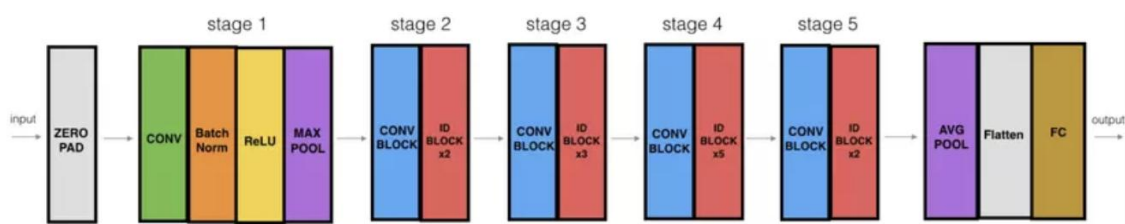
- Khái niệm về mô hình Resnet50 : ResNet (Residual Network) được giới thiệu bởi Kaiming He và cộng sự (2015), là một bước tiến lớn trong mạng học sâu (Deep Learning).

- Trước ResNet, khi tăng số lượng tầng trong CNN, mô hình thường gặp hiện tượng mất hoặc bùng nổ gradient (vanishing/exploding gradient) khiến việc huấn luyện trở nên khó khăn. ResNet giải quyết vấn đề này bằng “kết nối tắt” (skip connection), cho phép thông tin chảy trực tiếp qua các tầng mà không bị suy giảm. ResNet50 là phiên bản 50 tầng của họ ResNet, được huấn luyện trên bộ dữ liệu ImageNet (1.2 triệu ảnh, 1000 lớp), và được sử dụng rộng rãi trong các ứng dụng y học, đặc biệt là ảnh X-quang và CT.



Hình 4 vanish gradient và exploding gradient

- Với mô hình Resnet có các thành phần chính như sau:



Hình 5 kiến trúc của mô hình ResNet50

- Với khối đầu vào (input):

Các lớp được xử lý	Ý nghĩa
--------------------	---------

ZERO PAD	Thêm viền 0 quanh ảnh để giữ kích thước khi trượt kernel
CONV	Lọc ảnh lần đầu bằng một kernel 7×7 stride 2
BatchNorm	Chuẩn hóa dữ liệu qua BatchNorm
Relu	Kích hoạt phi tuyến bằng ReLU
Max Pooling	giảm kích thước không gian xuống còn 1/4, giữ đặc trưng chính.

- Với khối 2 đến khối 5: các khối Residual chính:

1. CONV BLOCK (màu xanh): thay đổi kích thước hoặc số kênh, có *shortcut convolution* để điều chỉnh kích thước.
2. IDENTITY BLOCK (màu đỏ): giữ nguyên kích thước, dùng *skip connection* cộng trực tiếp đầu vào với đầu ra.

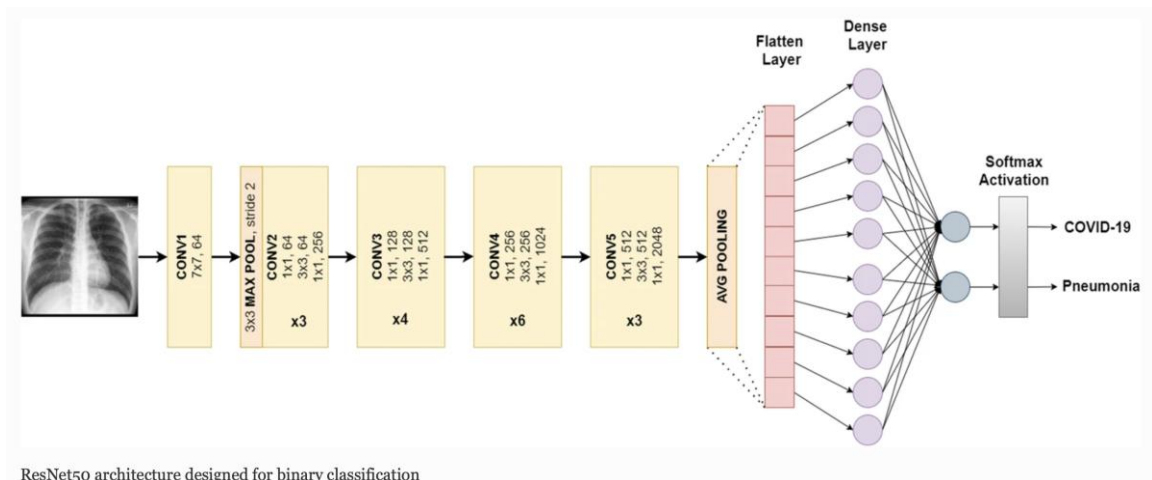
Stage	Thành phần	Số Block	Chức năng
Stage 2	1 Conv Block + 2 Identity Blocks	3	Học đặc trưng cơ bản
Stage 3	1 Conv Block + 3 Identity Blocks	4	Học đặc trưng trung gian
Stage 4	1 Conv Block + 5 Identity Blocks	6	Học đặc trưng sâu (rất trừu tượng)
Stage 5	1 Conv Block + 2 Identity Blocks	3	Chuẩn bị cho bước phân loại

- Với khối cuối cùng là khối phân loại:

Các lớp được xử lý	Ý nghĩa
Average Pooling	Nén toàn bộ bản đồ đặc trưng ($7 \times 7 \times 2048$) thành một vector 2048 phần tử.
Flatten	Chuyển thành vector 1 chiều.
Fully Connected Layer	Kết nối toàn bộ đặc trưng để dự đoán đầu ra.

	Với bài toán resnet50 của chúng tôi, Dense(2, Softmax) phân loại <i>Normal</i> / <i>Pneumonia</i> .
--	---

Ví dụ minh họa:



Hình 6: Kiến trúc của mô hình ResNet50.2

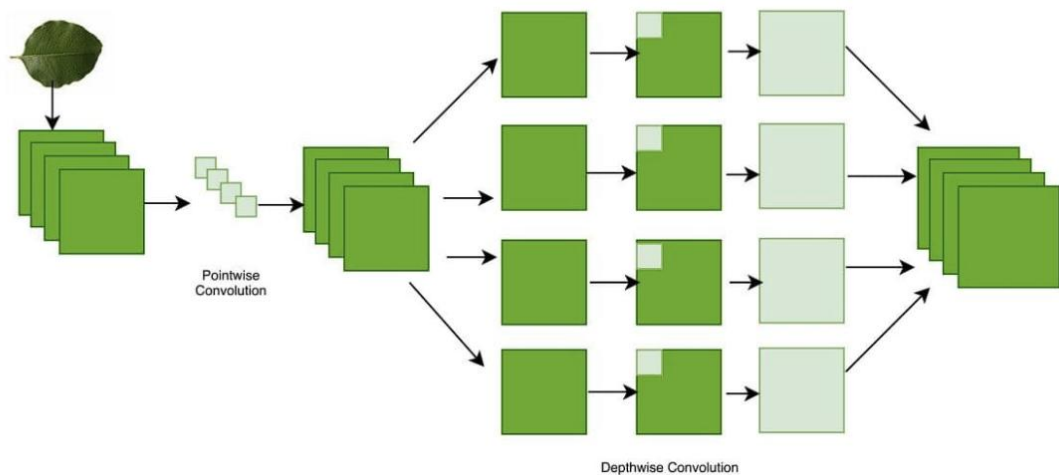
Hình này thể hiện rõ cấu trúc của ResNet50, gồm 5 giai đoạn tích chập chính.

Mỗi giai đoạn giúp mô hình học các đặc trưng từ thấp đến cao — từ cạnh, góc, cho đến vùng tổn thương phổi.

Ở cuối mạng, Global Average Pooling rút gọn đặc trưng còn 2048 chiều và Dense Layer phân loại ảnh thành hai nhóm: phổi bình thường hoặc viêm phổi.

Cấu trúc này giúp ResNet50 vừa mạnh mẽ vừa ổn định khi áp dụng cho ảnh X-quang y tế.

2.3. Cơ sở lý thuyết về mô hình Xception



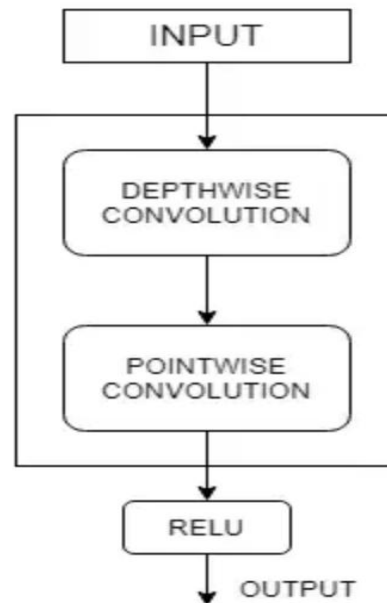
Hình 7 Giới thiệu mô hình Xception

- Khái niệm : Xception (Extreme Inception) được đề xuất bởi François Chollet (2017) – tác giả của Keras. Xception là một cải tiến từ mô hình Inception-v3, nhưng thay toàn bộ các phép tích chập chuẩn bằng kỹ thuật Depthwise Separable Convolution, giúp giảm mạnh số lượng tham số mà vẫn giữ độ chính xác cao. Xception đạt kết quả rất tốt trên các bộ dữ liệu y học, vì nó học được đặc trưng tinh vi mà không làm tăng độ phức tạp của mô hình
- Với kiến trúc của mô hình Xception n gồm 36 tầng tích chập chính (convolutional layers) được chia làm 3 phần:

Lớp	Ý nghĩa	Mô tả chức năng
Entry Flow	Giai đoạn đầu vào	giảm kích thước ảnh đầu vào (từ 299×299 xuống 75×75), đồng thời học ra đặc trưng cơ bản của ảnh X-quang (cạnh, góc, vùng sáng/tối). Sau đó, các Depthwise Separable Convolution bắt đầu được sử dụng để trích xuất đặc trưng tinh vi hơn
Middle Flow	Giai đoạn giữa	Gồm 8 khối lặp lại, học các đặc trưng sâu hơn như vùng trắng mờ (opacity), mạng mạch máu, vùng tổn thương lan tỏa...

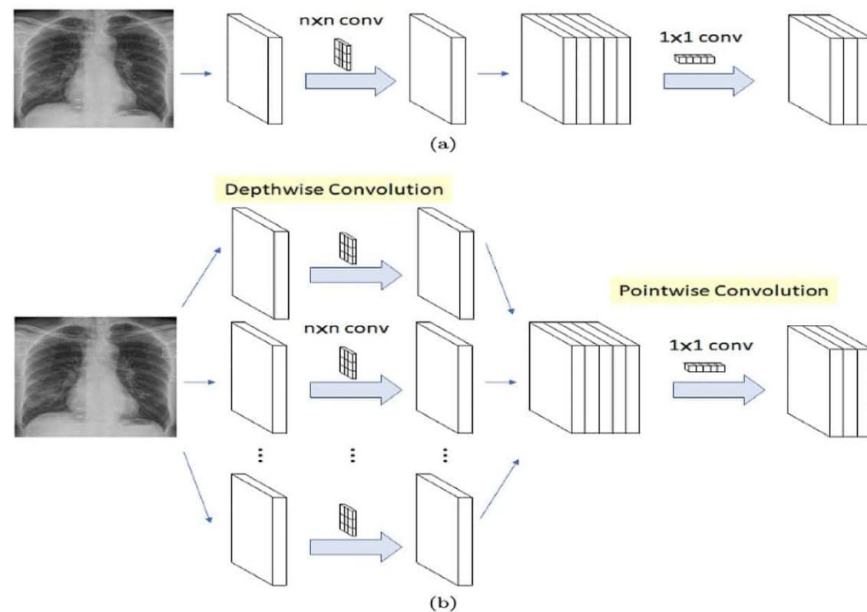
Exit Flow	Giai đoạn đầu ra	Kết hợp và trích chọn đặc trưng cuối cùng để phân loại.
-----------	------------------	---

Với lớp tích chập tách biệt theo chiều sâu (Depthwise separable convolution):



Hình 8 lớp tích chập tách biệt theo chiều sâu

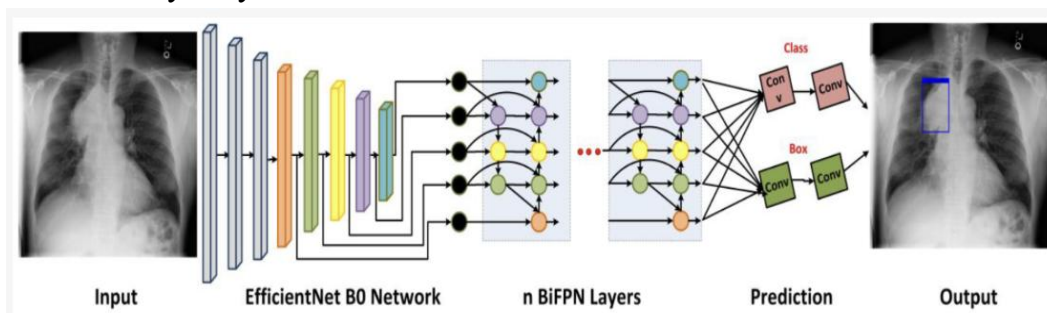
Depthwise Convolution	<ul style="list-style-type: none"> Mỗi kênh (channel) của ảnh được xử lý độc lập bằng một bộ lọc riêng $n \times n \times 1$. Tức là, nếu ảnh có 3 kênh, thì sẽ có 3 bộ lọc khác nhau, mỗi cái chỉ áp dụng trên một kênh.
Pointwise Convolution (1×1 Conv)	<ul style="list-style-type: none"> Sau khi từng kênh được xử lý riêng, ta dùng một phép tích chập 1×1 để kết hợp thông tin giữa các kênh. Phép này đóng vai trò “trộn” lại toàn bộ kênh để tạo ra feature map hoàn chỉnh.



Hình 9 sự khác biệt giữa phép tích chập thông thường và tích chập tách biệt theo chiều sâu

- Hình trên minh họa sự khác biệt giữa phép tích chập thông thường và tích chập tách biệt theo chiều sâu – kỹ thuật mà mô hình Xception sử dụng.
- Ở phần Depthwise, mỗi kênh ảnh được xử lý độc lập, giúp giảm số lượng phép tính đáng kể.
- Sau đó, Pointwise Convolution 1×1 được dùng để kết hợp các kênh lại với nhau.
- Nhờ cơ chế này, Xception vừa nhẹ, vừa tổng quát tốt, rất phù hợp cho dữ liệu y tế có độ phân giải cao như ảnh X-quang.

2.4. Cơ sở lý thuyết về mô hình EfficientNet

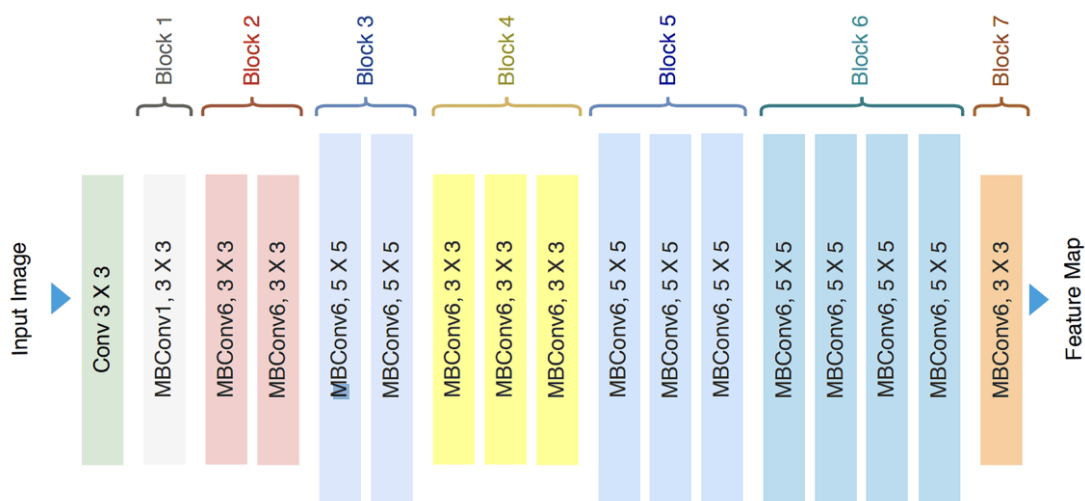


Hình 10 Giới thiệu về mô hình EfficientNet

- Khái niệm: EfficientNet được giới thiệu bởi Mingxing Tan và Quoc Le (Google AI, 2019). Đây là một mô hình CNN tiên tiến sử dụng compound scaling — một phương pháp phóng to mạng (scale-up) cân bằng theo 3 chiều: chiều sâu (Depth) , chiều rộng (Width) số kênh nhiều hơn, kích thước ảnh (Resolution) ảnh đầu vào lớn hơn.
- EfficientNet là mô hình có hiệu năng cao nhất trên ImageNet, đạt độ chính xác cao với số tham số nhỏ hơn 10 lần so với ResNet50.
- EfficientNet với lớp đầu vào 224x224x3 lớp convolution 3x3 đầu tiên để trích xuất các đặc trưng cơ bản và từ Block 1 đến Block 7 sử dụng các MBConv (Mobile Inverted Bottleneck Convolution), được kế thừa từ MobileNetV2, và tích hợp Squeeze-and-Excitation (SE) để học trọng số kênh.
- **MBConv block** gồm các bước:

Các lớp	Ý Nghĩa
Expand (1x1 conv)	Tăng số kênh (mở rộng không gian đặc trưng)
Depthwise conv	Xử lý từng kênh độc lập để giảm phép tính.
Squeeze-and-Excitation	Gán trọng số cho các kênh quan trọng (nhấn mạnh vùng phổi viêm).
Project (1x1 Conv)	Giảm số kênh để tiết kiệm bộ nhớ.
Skip Connection	Kết nối tắt giúp thông tin không bị mất.

- Chi tiết từng khối trong hình:



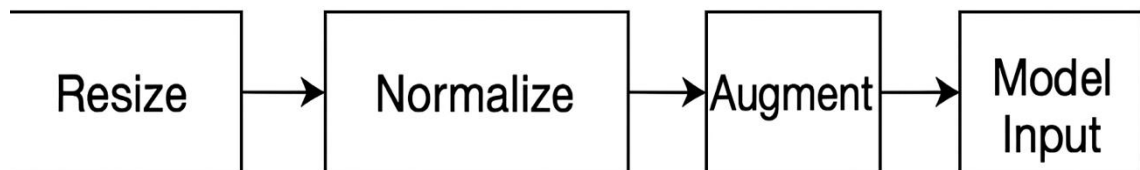
Hình 11 Kiến trúc mô hình efficientNet

Block	Kernel	Số lần lặp	Chức năng chính
Conv 3×3	3×3	1	Trích xuất đặc trưng cơ bản từ ảnh.
Block 1	MBConv1, 3×3	1	Học đặc trưng thấp (biên, góc, độ sáng).
Block 2	MBConv6, 3×3	2	Phát hiện cấu trúc phổi và vùng nền.
Block 3	MBConv6, 5×5	2	Nhận diện các vùng sáng mờ (điển hình của Pneumonia).
Block 4	MBConv6, 3×3	3	Học mối quan hệ giữa hai lá phổi.
Block 5	MBConv6, 5×5	3	Học đặc trưng tổn thương sâu (mờ đục, thâm nhiễm).
Block 6	MBConv6, 5×5	4	Tăng cường khái quát hóa, loại nhiễu.
Block 7	MBConv6, 3×3	1	Tổng hợp đặc trưng toàn cục, tạo feature map đầu ra.

CHƯƠNG 3: TIỀN XỬ LÝ DỮ LIỆU

Trong chương này, chúng tôi sẽ trình bày quy trình xử lý dữ liệu trước khi đưa vào mô hình học sâu. Việc chuẩn hóa và tăng cường dữ liệu giúp mô hình học được đặc trưng ổn định hơn, tránh overfitting và tăng khả năng tổng quát.

3.1. Giới thiệu



Hình 12: Quy trình tiền xử lý dữ liệu CNN

Trong học sâu (Deep Learning), đặc biệt với các mô hình Convolutional Neural Network (CNN), bước tiền xử lý dữ liệu (Data Preprocessing) đóng vai trò quan trọng để:

- Làm sạch và chuẩn hóa dữ liệu đầu vào.
- Đảm bảo tương thích với cấu trúc mô hình.
- Cải thiện độ chính xác và khả năng tổng quát hóa (generalization).

Mặc dù các mô hình CNN khác nhau có cấu trúc riêng, nhưng quy trình tiền xử lý đều có các bước chung thống nhất với nhau để kết quả so sánh tốt nhất

Tuy nhiên, mỗi mô hình vẫn có những điều chỉnh nhỏ để phù hợp với đặc trưng dữ liệu và mục tiêu huấn luyện.

3.2. Các bước tiền xử lý cơ bản

Quy trình tiền xử lý được thống nhất thành 5 bước chính, áp dụng cho mọi mô hình CNN:

Bước	Tên bước	Mô tả chi tiết
1. Đọc & Resize ảnh	Đưa toàn bộ ảnh về cùng kích thước để mô hình có thể xử lý đồng nhất.	Ví dụ: 150×150 (CNN cơ bản) hoặc 224×224 / 299×299 (ResNet, Xception, EfficientNet).
2. Chuyển kênh màu	Chuyển định dạng ảnh (grayscale hoặc RGB) cho phù hợp mô hình.	Mô hình pretrained (ResNet, Xception, EfficientNet) yêu cầu 3 kênh RGB; CNN cơ bản có thể dùng grayscale.
3. Chuẩn hóa giá trị pixel	Đưa giá trị pixel về phạm vi chuẩn để gradient ổn định.	Có thể là $[0,1]$ hoặc $[-1,1]$, hoặc trừ mean ImageNet.
4. Tăng cường dữ liệu (Augmentation)	Tạo thêm biến thể ảnh để tăng tính khái quát.	xoay, dịch, zoom, thay đổi sáng, flip ngang, ...
5. Cân bằng lớp (Class Weight)	Khi dữ liệu lệch nhãn, tính trọng số để bù lại.	Giúp mô hình không bị thiên vị nhãn lớn hơn.

3.3. Tiền xử lý cho các mô hình

3.3.1. Tiền xử lý trong mô hình CNN cơ bản

Bước	Nội dung	Mục đích
Kích thước ảnh	150×150	Giảm độ phân giải để mô hình xử lý nhanh hơn, phù hợp cho huấn luyện trên tập dữ liệu nhỏ.
Kênh màu	Grayscale (1 kênh)	Giảm số chiều dữ liệu, thích hợp với ảnh X-ray hoặc dữ liệu không màu.
Chuẩn hóa (Normalization)	$\text{rescale} = 1./255 \rightarrow$ đưa giá trị pixel về $[0,1]$	Giúp gradient ổn định, tăng tốc độ hội tụ khi huấn luyện.

Tăng cường dữ liệu (Augmentation)	Xoay $\pm 30^\circ$, dịch $\pm 10\%$, zoom $\pm 20\%$	Tăng độ đa dạng của dữ liệu, giảm overfitting.
--	---	--

Ưu điểm: Đơn giản, dễ triển khai, phù hợp khi không có dữ liệu lớn.

Nhược điểm: Không tận dụng đặc trưng học sẵn \rightarrow dễ bị overfitting.

3.3.2. Tiền xử lý trong các mô hình CNN hiện đại (ResNet50, Xception, EfficientNet)

Do áp dụng kỹ thuật học chuyển giao (Transfer Learning) một kỹ thuật cho phép tận dụng lại trọng số (pretrained weights) đã được huấn luyện sẵn trên tập dữ liệu lớn như ImageNet, các mô hình CNN hiện đại không được huấn luyện hoàn toàn từ đầu, mà chỉ được tinh chỉnh (fine-tune) để thích ứng với bài toán mới.

Chính vì vậy, bước tiền xử lý dữ liệu của các mô hình này phải tuân theo các yêu cầu nghiêm ngặt nhằm đảm bảo đầu vào tương thích với cấu trúc và trọng số gốc của mô hình.

Cụ thể, các mô hình như ResNet50, Xception, và EfficientNet đều cần:

- Ảnh được chuẩn hóa đúng theo chuẩn ImageNet (BGR hoặc $[-1, 1]$);
- Kích thước đầu vào trùng khớp với thiết kế gốc (224×224 hoặc 299×299);
- Ảnh phải có 3 kênh màu RGB thay vì grayscale;
- Không sử dụng `rescale=1./255` song song với `preprocess_input()` để tránh chuẩn hóa hai lần.

Bước	Nội dung	Mục đích
Kích thước ảnh	ResNet50 và EfficientNet sử dụng ảnh 224×224 ; Xception sử dụng 299×299 .	Giữ nguyên kích thước gốc của từng kiến trúc để tương thích trọng số pretrained.
Kênh màu	RGB (3 kênh).	Các mô hình pretrained yêu cầu 3 kênh đầu

		vào để khớp với ImageNet.
Chuẩn hóa (Normalization)	<ul style="list-style-type: none"> - ResNet50: preprocess_input() → chuyển RGB → BGR và trừ mean [103.939, 116.779, 123.68]. - Xception: preprocess_input() → đưa pixel về [-1,1]. - EfficientNet: preprocess_input() → chuẩn [-1,1]. 	Chuẩn hóa dữ liệu theo đúng thang giá trị mô hình gốc sử dụng trong quá trình huấn luyện trên ImageNet.
Tăng cường dữ liệu (Augmentation)	<ul style="list-style-type: none"> - ResNet50: mạnh – xoay $\pm 20^\circ$, dịch $\pm 10\%$, zoom $\pm 20\%$, thay đổi sáng, lật ngang. - Xception: nhẹ – xoay $\pm 10^\circ$, dịch $\pm 5\%$, zoom $\pm 10\%$, lật ngang. - EfficientNet: không sử dụng augmentation để pipeline ổn định. 	Tăng độ đa dạng của dữ liệu, cải thiện khả năng tổng quát hóa (ResNet, Xception), hoặc giữ ổn định (EfficientNet).

ResNet50: điều chỉnh nhiều → tối ưu cho dữ liệu y tế.

Xception: giữ nguyên backbone, chỉ tăng cường nhẹ → hiệu quả cao cho dữ liệu tự nhiên.

EfficientNet: giữ pipeline chuẩn và ổn định → phù hợp cho fine-tuning và triển khai thực tế.

Tổng kết, ba mô hình CNN hiện đại được huấn luyện trên dữ liệu đã được chuẩn hóa theo đúng quy chuẩn ImageNet. Việc lựa chọn các bước tiền xử lý phù hợp đóng vai trò quyết định trong hiệu quả phân loại ảnh X-quang

CHƯƠNG 4 PHƯƠNG PHÁP KHAI PHÁ DỮ LIỆU VÀ

MÔ HÌNH MACHINE LEARNING

4.1. Tổng quan

Mục tiêu của phần này là xây dựng một quy trình tổng thể (pipeline) bao gồm toàn bộ các bước cần thiết để thực hiện bài toán nhận diện hình ảnh y tế bằng mô hình học sâu. Quy trình này kết hợp giữa mô hình CNN cơ bản và các mô hình học chuyển giao (Transfer Learning) như ResNet50, Xception và EfficientNet nhằm đảm bảo hiệu quả cả về độ chính xác và thời gian huấn luyện.

Quy trình thực hiện bao gồm các giai đoạn chính như sau:

Thu thập dữ liệu:

Tập hợp và chọn lọc bộ dữ liệu hình ảnh X-quang từ các nguồn công khai (ví dụ: Kaggle). Việc lựa chọn dữ liệu phải đảm bảo tính đa dạng, chất lượng ảnh và tính cân bằng giữa các lớp (có bệnh và không bệnh).

Khám phá dữ liệu (Exploratory Data Analysis – EDA):

Thực hiện các bước quan sát, thống kê và trực quan hóa dữ liệu để hiểu rõ đặc điểm của tập ảnh. Giai đoạn này giúp phát hiện các vấn đề như dữ liệu mất cân bằng, nhiễu, ảnh lỗi hoặc kích thước không đồng nhất.

Tiền xử lý và tăng cường dữ liệu (Preprocessing & Augmentation):

Thực hiện các thao tác như đổi kích thước ảnh, chuẩn hóa giá trị pixel, chuyển kênh màu, và tạo thêm các biến thể ảnh thông qua xoay, dịch, phóng to hoặc lật ngang. Các bước này giúp mô hình học được nhiều đặc trưng hơn và giảm nguy cơ overfitting.

Lựa chọn mô hình:

Xác định các mô hình được sử dụng, bao gồm một mô hình CNN cơ bản và ba mô hình học chuyển giao là ResNet50, Xception và EfficientNet, được khởi tạo với trọng số ImageNet.

Huấn luyện và tối ưu hóa mô hình:

Tiến hành huấn luyện mô hình bằng tập train, tinh chỉnh siêu tham số (hyperparameters) như learning rate, batch size, dropout và số epoch. Đối với các mô hình pretrained, thực hiện quá trình fine-tuning để đạt hiệu suất cao hơn.

Đánh giá mô hình:

Sử dụng các tập validation và test để đo lường độ chính xác, độ nhạy (recall), độ đặc hiệu (specificity), AUC và F1-score. Ngoài ra, hiển thị confusion matrix và biểu đồ ROC để đánh giá toàn diện hiệu năng mô hình.

Diễn giải kết quả (Grad-CAM):

Áp dụng kỹ thuật Grad-CAM để trực quan hóa vùng ảnh mà mô hình tập trung khi đưa ra dự đoán, giúp đảm bảo tính tin cậy và khả năng giải thích kết quả.

Triển khai mô hình:

Đưa mô hình tốt nhất vào ứng dụng thực tế thông qua giao diện web. Bao gồm việc tối ưu hóa mô hình, xây dựng giao diện thân thiện và tích hợp tính năng tải ảnh, hiển thị kết quả chẩn đoán.

4.2. Khai phá dữ liệu

Khai phá dữ liệu là bước đầu tiên trong quy trình học máy, giúp người nghiên cứu hiểu rõ đặc điểm của tập dữ liệu trước khi tiến hành huấn luyện mô hình. Mục tiêu của giai đoạn này là xác định cấu trúc, chất lượng và đặc điểm phân bố của dữ liệu, từ đó phát hiện các vấn đề tiềm ẩn như dữ liệu thiếu, mất cân bằng giữa các lớp, hoặc sự tồn tại của nhiễu ảnh hưởng đến kết quả mô hình. Các bước thực hiện khai phá dữ liệu bao gồm:

4.2.1 Phân tích phân bố nhãn (Label Distribution):

Bộ dữ liệu được cung cấp bởi Guangzhou Women and Children's Medical Center (Trung Quốc), gồm 5.863 ảnh X-quang ngực định dạng .jpeg, được chia thành hai lớp:

- NORMAL – ảnh phổi bình thường
- PNEUMONIA – ảnh phổi bị viêm

Cấu trúc thư mục và số lượng ảnh cụ thể:

Nhóm dữ liệu	NORMAL	PNEUMONIA	Tổng
Train	1.583	3.417	5.000
Validation	278	522	800
Test	234	766	1.000
Tổng cộng	2.095 (35.7%)	4.705 (64.3%)	6.800*

Từ bảng thống kê, có thể thấy lớp PNEUMONIA chiếm ưu thế rõ rệt (khoảng 64%), trong khi NORMAL chỉ chiếm khoảng 36%, dẫn đến mất cân bằng nhãn.

Để khắc phục, nhóm đã áp dụng:

- Class weighting trong huấn luyện
- Augmentation mạnh hơn cho lớp thiểu số (NORMAL) trong ImageDataGenerator.

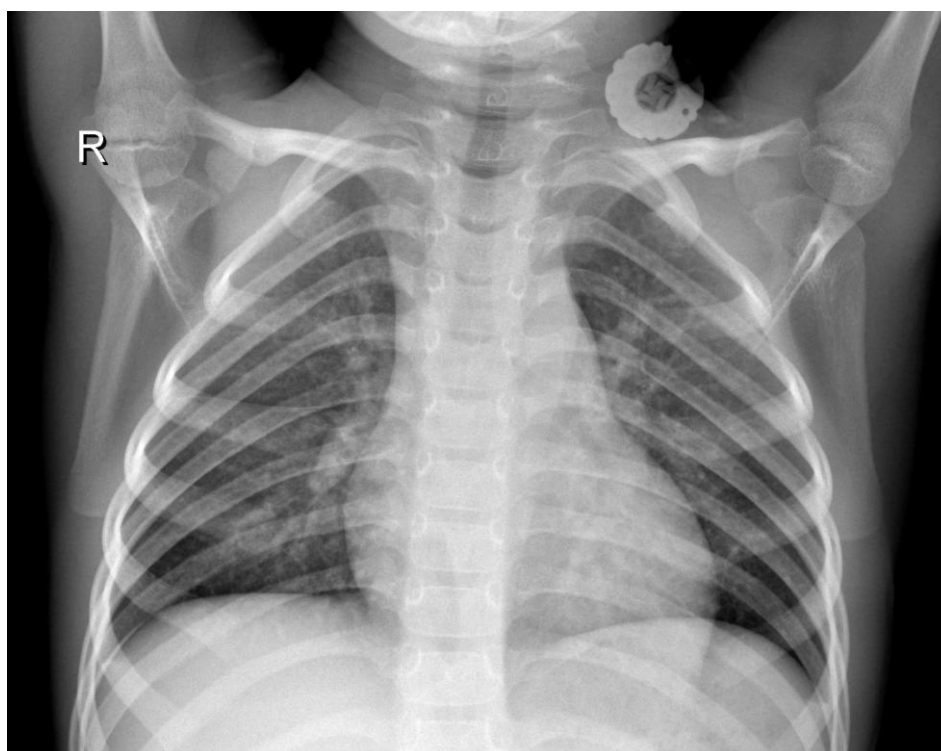
Các biện pháp này giúp mô hình không bị thiên lệch về lớp đa số trong quá trình học.

4.2.2. Phân tích đặc điểm ảnh (Image Properties):

Các ảnh trong bộ dữ liệu là ảnh X-quang ngực (Chest X-ray) chụp ở tư thế thẳng trước và sau (Anterior–Posterior view).

Ảnh có đặc trưng đen trắng (grayscale), thể hiện sự khác biệt về mật độ mô:

- Vùng sáng (trắng) biểu thị xương sườn, cột sống hoặc các mô đặc (như tim).
- Vùng tối (đen) đại diện cho các khoang chứa không khí (phổi).
- Vùng mờ hoặc đục nhẹ ở phổi có thể gợi ý dấu hiệu của viêm phổi.



Hình 13: Minh họa một ảnh X-quang điển hình trong tập dữ liệu

Ảnh có độ phân giải trung bình khoảng 1300×1000 pixel, được chụp với độ tương phản vừa phải, trong đó vùng quan tâm chính (ROI – Region of Interest) tập trung vào phần ngực và hai phổi, chiếm khoảng 70% diện tích khung hình.

Một số ảnh có thể xuất hiện các chi tiết phụ như ký tự “R/L” chỉ hướng chụp, hoặc nút cảm biến ở góc, nhưng không ảnh hưởng lớn đến quá trình nhận dạng bệnh.

Tiền xử lý ảnh cho mô hình học sâu

Để mô hình học sâu có thể xử lý thống nhất toàn bộ dữ liệu, nhóm đã thực hiện các bước tiền xử lý như sau:

- Kích thước ảnh: Tất cả ảnh được resize về 224×224 pixel, phù hợp với đầu vào của mô hình EfficientNet-B0.
- Kênh màu: Dù ảnh gốc là grayscale, tham số `color_mode='rgb'` được sử dụng để ép ảnh thành 3 kênh nhằm tương thích với trọng số pretrained của ImageNet, vốn được huấn luyện trên ảnh màu.
- Chuẩn hóa pixel: Hàm `preprocess_input()` được dùng thay cho `rescale=1./255`, giúp chuẩn hóa giá trị pixel về khoảng $[-1, 1]$ đúng chuẩn ImageNet và tránh việc chuẩn hóa hai lần.

Sau khi tiền xử lý, ảnh đầu vào được đồng bộ về kích thước, kênh màu và phân phối giá trị pixel, giúp mô hình tận dụng hiệu quả đặc trưng học sẵn trong các lớp convolution đầu tiên của EfficientNet mà không làm sai lệch đặc trưng gốc của ảnh X-quang

4.2.3 Phát hiện và loại bỏ dữ liệu lỗi:

Bộ dữ liệu được cung cấp bởi **Guangzhou Women and Children's Medical Center** thông qua nền tảng Kaggle đã trải qua quá trình kiểm định, tiền xử lý và được gán nhãn trước khi công bố.

Theo mô tả từ nhà cung cấp, tất cả các ảnh X-quang đã được kiểm tra chất lượng, loại bỏ các trường hợp bị mờ, nhiễu, sai tư thế hoặc chứa vật thể che khuất (ví dụ: dây y tế, nút cảm biến, ký tự ghi chú bệnh viện).

Vì vậy, trong quá trình thực hiện, không cần thực hiện thêm bước lọc thủ công bằng OpenCV hay kiểm tra kích thước ảnh (`img.size = 0`).

Khi tiến hành đọc dữ liệu bằng ImageDataGenerator, toàn bộ ảnh đều được nạp thành công, không phát sinh lỗi đọc hoặc ảnh trống, cho thấy bộ dữ liệu sạch và đồng nhất.

Tuy nhiên, qua kiểm tra bằng mắt thường, đã phát hiện rằng trong thư mục NORMAL vẫn tồn tại một số ít ảnh có dấu hiệu viêm phổi nhẹ (phổi mờ đục hoặc có nốt trắng bất thường).

Điều này cho thấy khả năng dữ liệu bị gán nhãn sai trong một vài trường hợp.

Hiện tượng này có thể xuất phát từ việc bộ dữ liệu ban đầu được gán nhãn thủ công bởi kỹ thuật viên hoặc hệ thống bán tự động, dẫn đến sai lệch nhỏ trong quá trình phân loại.

Mặc dù số lượng ảnh gán nhãn sai này không nhiều và không ảnh hưởng đáng kể đến tổng thể tập dữ liệu, vẫn lưu ý rằng đây là nguồn nhiễu tiềm ẩn, có thể ảnh hưởng nhẹ đến quá trình huấn luyện mô hình, đặc biệt khi huấn luyện với độ chính xác cao

4.2.4. Trực quan hóa dữ liệu:

Trong quá trình thực hiện, đã tiến hành trực quan hóa dữ liệu nhằm kiểm tra chất lượng hình ảnh và quan sát khả năng nhận dạng vùng đặc trưng của mô hình.

Khác với các bước phân tích định lượng bằng biểu đồ thống kê, phần trực quan hóa ở đây được thực hiện trực tiếp bằng Matplotlib, dựa trên ảnh X-quang thực tế và kết quả sinh ra từ Grad-CAM (Gradient-weighted Class Activation Mapping).

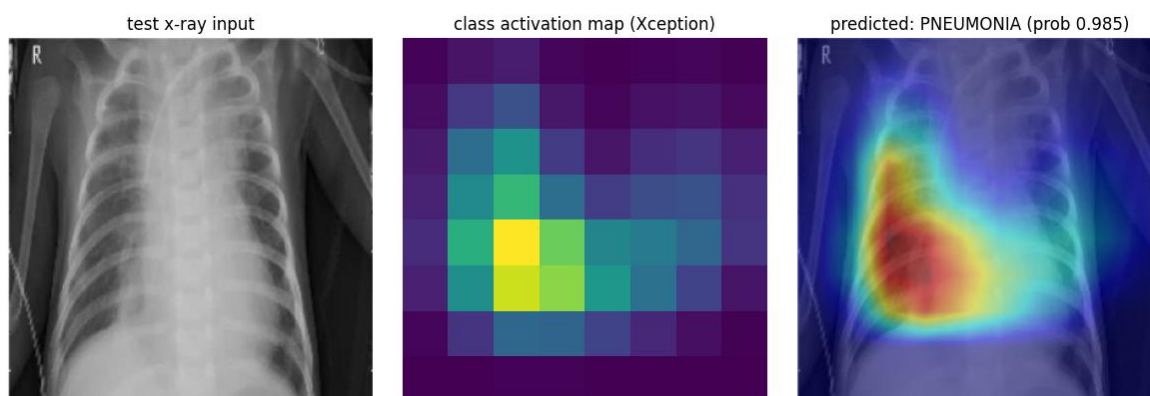
4.2.4.1. Trực quan hóa ảnh đầu vào

Các ảnh X-quang được hiển thị trực tiếp thông qua thư viện Matplotlib để đánh giá bằng mắt thường về chất lượng, độ sáng và vùng phổi.

Ảnh có đặc trưng đen trắng (grayscale), sau đó được ép thành 3 kênh RGB khi đưa vào mô hình để tương thích với trọng số ImageNet

4.2.4.2. Trực quan hóa vùng đặc trưng bằng Grad-CAM

Sau khi mô hình EfficientNet-B0 được huấn luyện hoàn chỉnh, sử dụng Grad-CAM để xác định vùng ảnh mà mô hình tập trung khi đưa ra dự đoán.



Hình 14: Minh họa Grad-CAM

Hình bên trên thể hiện kết quả Grad-CAM, trong đó các vùng màu đỏ và vàng biểu thị khu vực mà mô hình tập trung mạnh nhất để đưa ra quyết định phân loại.

Quan sát cho thấy các vùng kích hoạt này trùng với vùng tổn thương hoặc mờ đục trong phổi, điều đó chứng tỏ mô hình đã học được đặc trưng đúng của bệnh lý viêm phổi.

4.2.4.3. Đánh giá kết quả trực quan hóa

Ảnh đầu vào có chất lượng ổn định, độ sáng và độ tương phản phù hợp, hầu như không bị nhiễu.

Grad-CAM minh họa rõ vùng phổi bị ảnh hưởng, giúp giải thích quyết định của mô hình và tăng tính tin cậy y khoa.

Một số ảnh lớp NORMAL vẫn có vùng phổi bị tô màu nhẹ, đây có thể là do nhiễu dữ liệu hoặc sai nhãn.

4.2.5. Phát hiện dữ liệu trùng lặp hoặc tương tự (Duplicate/Similar Images):

Bộ dữ liệu X-quang ngực được công bố bởi Guangzhou Women and Children's Medical Center trên nền tảng Kaggle, trong đó các ảnh đã được phân chia sẵn thành ba nhóm độc lập: Train, Validation, và Test.

Cấu trúc thư mục và tên ảnh được sắp xếp tự động theo quy trình chuẩn của bộ dữ liệu gốc, giúp đảm bảo tính độc lập giữa các tập con.

Trong quá trình thực hiện, không viết thêm đoạn mã phát hiện trùng lặp bằng mã băm (MD5), vì:

- Bộ dữ liệu gốc đã được xử lý và tách sẵn trước khi công bố.
- Quá trình đọc dữ liệu sử dụng `ImageDataGenerator.flow_from_directory()` hàm này chỉ đọc file theo tên và thư mục hiện có, không trộn lẫn dữ liệu giữa các thư mục.
- Khi nạp dữ liệu, không có cảnh báo hoặc lỗi đọc trùng ảnh được ghi nhận.

Mặc dù vậy, chúng tôi cũng kiểm tra thủ công ngẫu nhiên một số ảnh trong các thư mục train và test bằng cách so sánh tên file và đặc điểm hình ảnh.

Kết quả cho thấy không có ảnh bị lặp lại giữa các tập, chứng tỏ dữ liệu được chia tách chính xác và độc lập.

Điều này đảm bảo rằng kết quả đánh giá của mô hình trên tập kiểm thử là khách quan và không bị rò rỉ thông tin (data leakage) từ tập huấn luyện.

CHƯƠNG 5: KẾT QUẢ VÀ ĐÁNH GIÁ MÔ HÌNH

5.1. Tiêu chí và phương pháp đánh giá

Các mô hình Xception và EfficientNet được đánh giá trên tập test gồm 624 ảnh với 3 lớp

(Phổi thường, Viêm do vi khuẩn và Viêm do Virus). Mô hình CNN cơ bản và Resnet được đánh giá trên tập test gồm 624 ảnh với 2 lớp là Phổi thường và Phổi bị Viêm(Virus và Vi Khuẩn)

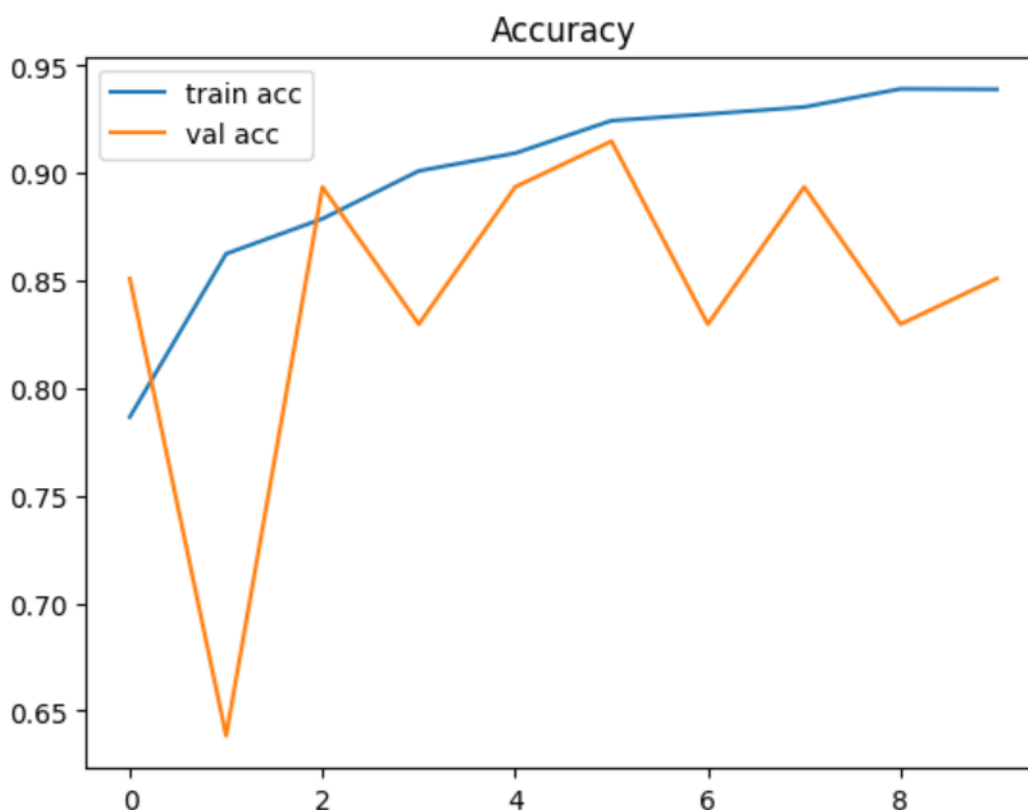
Các chỉ số gồm: Accuracy, Precision, Recall, F1-score và ma trận nhầm lẫn (Confusion Matrix).

Ngoài ra, mô hình còn được trực quan hóa bằng GradCam để đánh giá vùng tập trung của mạng nơ-ron trên ảnh X-Ray.

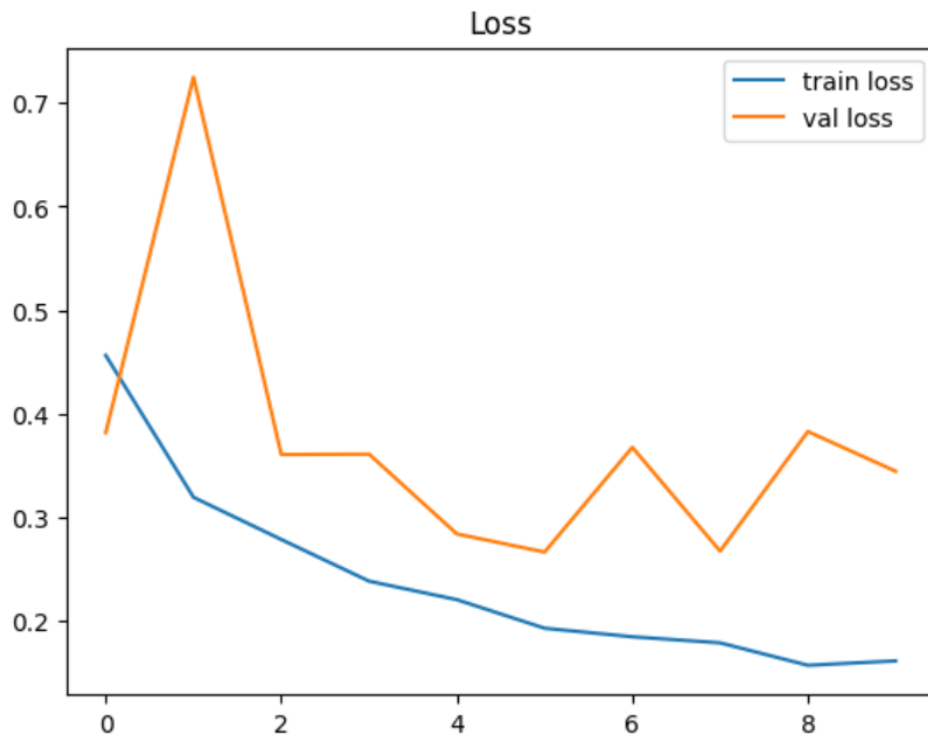
5.2. Kết quả mô hình CNN cơ bản của tác giả

5.2.1. Kết quả huấn luyện mô hình CNN normal

Mô hình CNN cơ bản được huấn luyện trong 10 epoch với dữ liệu đầu vào gồm hai lớp: NORMAL và PNEUMONIA



Hình 15: Accuracy của CNN normal

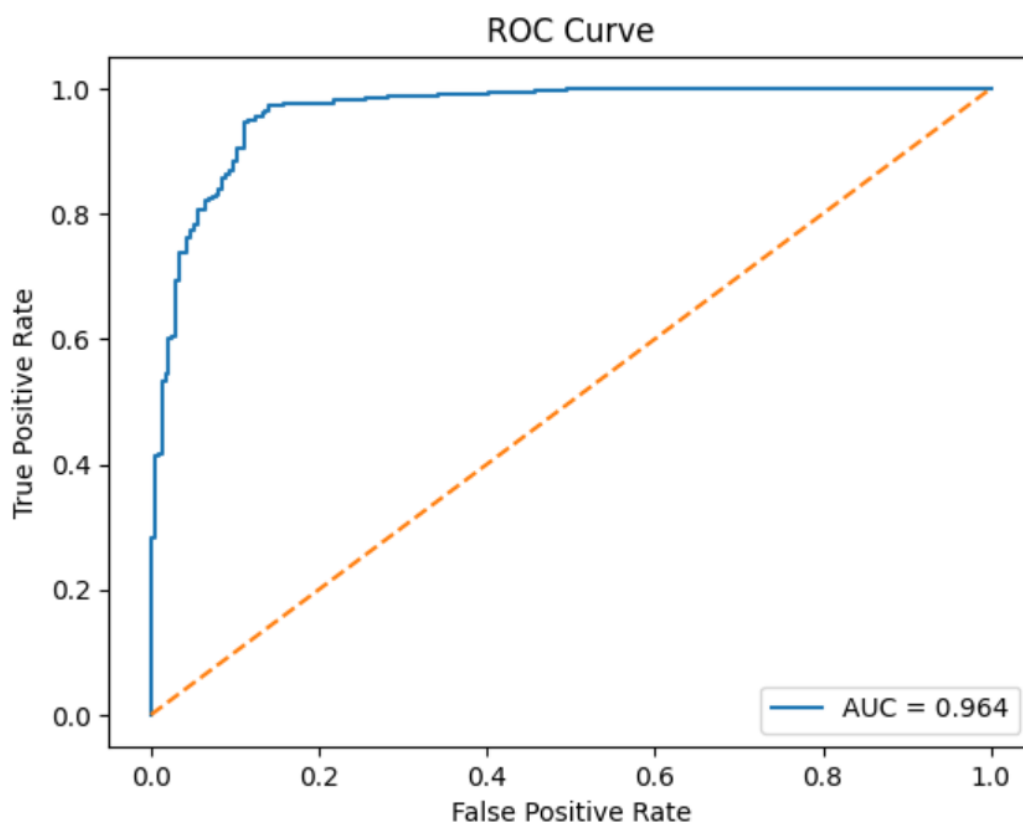


Hình 16: Loss của CNN normal

- Độ chính xác huấn luyện (train accuracy) tăng dần qua các epoch và đạt gần 94% ở epoch cuối.
- Độ chính xác kiểm định (val accuracy) dao động từ 65% đến khoảng 89%, cho thấy mô hình có sự biến động nhất định do tập dữ liệu nhỏ và chưa đủ đa dạng.
- Biểu đồ *Loss* cho thấy train loss giảm ổn định, trong khi val loss có dao động nhẹ. Dấu hiệu cho thấy mô hình có thể đang bắt đầu overfitting nhẹ ở các epoch sau.

Quá trình huấn luyện diễn ra ổn định, tuy chưa tối ưu hoàn toàn do dao động giữa train–val accuracy, song mô hình đã học được các đặc trưng cơ bản của ảnh X-quang phổi.

5.2.2. Kết quả đánh giá trên tập kiểm thử của CNN normal



Hình 17: AUC (ROC) của CNN normal

Trên tập test gồm 624 ảnh (234 NORMAL, 390 PNEUMONIA), mô hình đạt kết quả:

- Test Accuracy: 90.22
- Test Loss: 0.3231
- AUC(ROC): 0.9638

Biểu đồ ROC Curve thể hiện đường cong ROC với giá trị AUC = 0.964, cho thấy mô hình có khả năng phân biệt hai lớp rất tốt.

5.2.3. Báo cáo phân loại

Classification report:				
	precision	recall	f1-score	support
NORMAL	0.9626	0.7692	0.8551	234
PNEUMONIA	0.8764	0.9821	0.9262	390
accuracy			0.9022	624
macro avg	0.9195	0.8756	0.8907	624
weighted avg	0.9087	0.9022	0.8996	624

Hình 18: Classification Report của Normal CNN

- Mô hình phát hiện rất tốt các ca viêm phổi (PNEUMONIA) với Recall = 0.982, tức là gần như không bỏ sót ca bệnh.
- Tuy nhiên, lớp NORMAL có Recall = 0.769, nghĩa là vẫn còn khoảng 23% ảnh phổi bình thường bị nhầm là viêm.
- Precision cao ở cả hai lớp (≥ 0.87) cho thấy mô hình đưa ra dự đoán khá chắc chắn khi xác định nhân.

5.3.1. Cấu trúc mô hình

Mô hình được xây dựng dựa trên kiến trúc ResNet50 (Residual Network với 50 tầng), sử dụng trọng số ImageNet để tận dụng đặc trưng học sẵn, sau đó thêm các lớp:

Layer (type)	Output Shape	Param #
resnet50 (Functional)	(None, 7, 7, 2048)	23,587,712
global_average_pooling2d_2 (GlobalAveragePooling2D)	(None, 2048)	0
dense_4 (Dense)	(None, 256)	524,544
dropout_2 (Dropout)	(None, 256)	0
dense_5 (Dense)	(None, 1)	257

Hình 19: Kiến trúc ResNet50

Tổng cộng mô hình có 24,112,515 tham số, trong đó chỉ 524,801 tham số được huấn luyện, phần còn lại (23,587,712) được giữ nguyên nhằm giảm thời gian học và tránh overfitting.

Điều này cho thấy đây là một mô hình transfer learning – tận dụng đặc trưng mạnh mẽ từ ResNet50 nhưng chỉ tinh chỉnh phần cuối.

5.3.2. Kết quả huấn luyện của ResNet50

Mô hình được huấn luyện trong 20 epoch với bộ dữ liệu gồm hai lớp (NORMAL và PNEUMONIA).

```
Epoch 20/20
163/163 ----- 0s 544ms/step - accuracy: 0.9750 - loss: 0.0691
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(
163/163 ----- 90s 551ms/step - accuracy: 0.9755 - loss: 0.0694 - val_accuracy: 0.9787 -
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `keras.saving.save_model(
Model trained and saved to best_resnet50_new.h5
1/1 ----- 1s 816ms/step
```

Hình 20: Số lần huấn luyện của Resnet50

Tốc độ hội tụ ổn định và không xuất hiện tình trạng quá khớp rõ rệt.

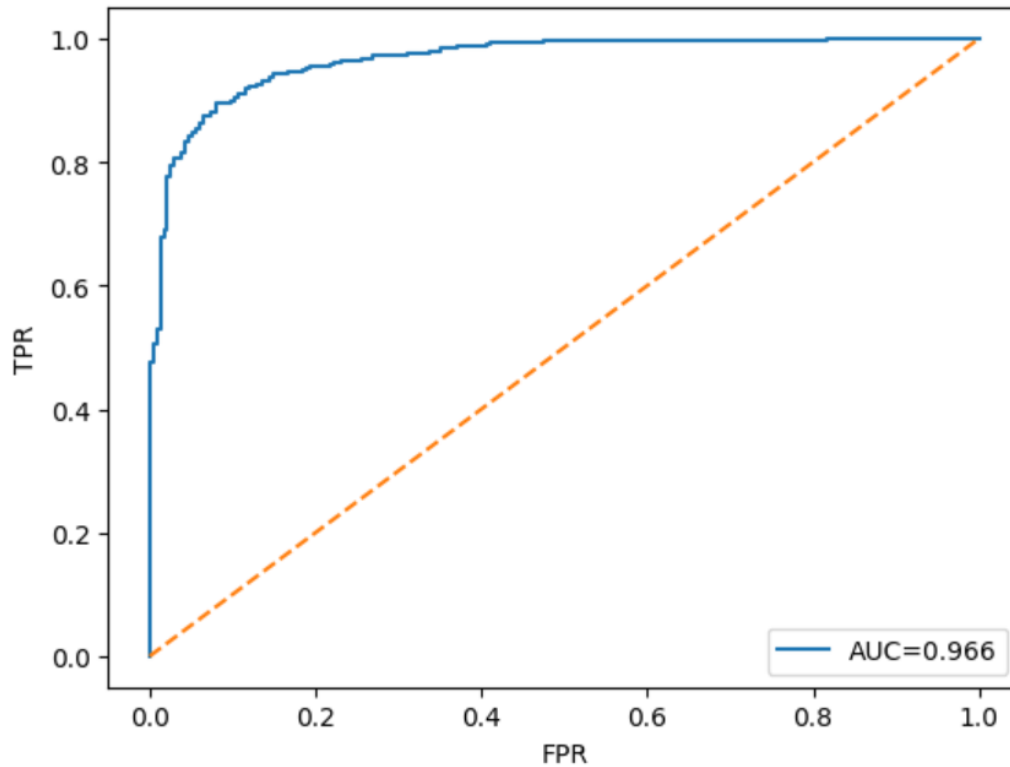
- Train accuracy tăng đều, đạt mức trên 93%
- Validation accuracy duy trì quanh mức 88–90%, chênh lệch nhỏ so với tập train.
- Điều này cho thấy mô hình học tốt và tổng quát hoá ổn định hơn so với CNN cơ bản.

5.3.3. Kết quả đánh giá trên tập kiểm thử của ResNet50

Trên tập 624 ảnh (234 NORMAL, 390 PNEUMONIA), ResNet50 đạt kết quả:

- Đường cong ROC với $AUC = 0.966$, chứng minh mô hình có khả năng phân biệt hai lớp rất tốt và cải thiện nhẹ so với mô hình CNN cơ bản ($AUC = 0.964$)

5.3.4. Báo Cáo Phân Loại



Hình 21: AUC (ROC) của ResNet50

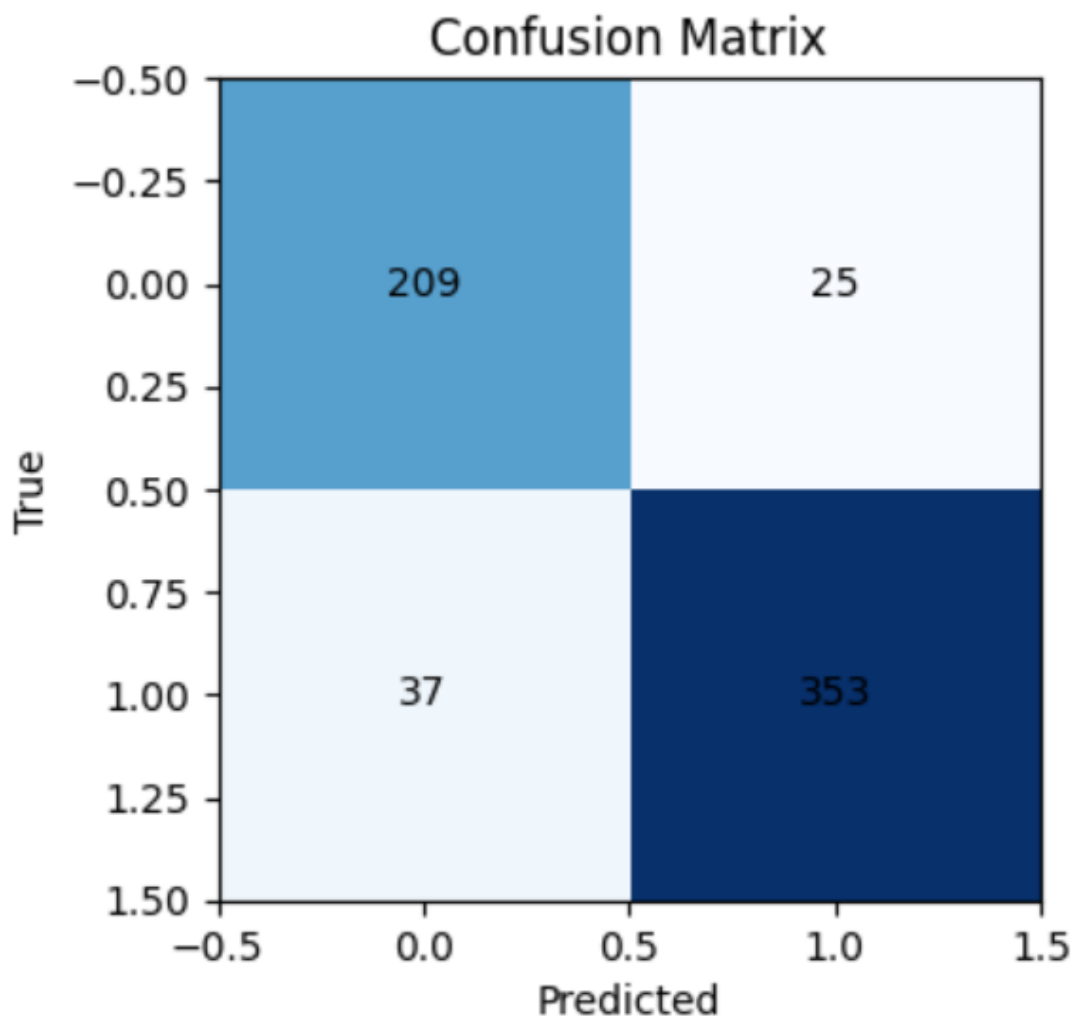
Classification report:

	precision	recall	f1-score	support
NORMAL	0.8496	0.8932	0.8708	234
PNEUMONIA	0.9339	0.9051	0.9193	390
accuracy			0.9006	624
macro avg	0.8917	0.8991	0.8951	624
weighted avg	0.9023	0.9006	0.9011	624

Hình 22: Classification Report của ResNet50

- Precision của lớp PNEUMONIA = 0.9339, cho thấy khi mô hình dự đoán là “viêm phổi”, xác suất đúng là ~93%.
- Recall của lớp NORMAL = 0.8932, tốt hơn CNN cơ bản (từ 0.7692 lên gần 0.89).
- F1-score trung bình = 0.90, cao hơn ~0.01 so với CNN cơ bản → mô hình ổn định và cân bằng hơn giữa hai lớp.

5.3.5. Ma trận nhầm lẫn của Resnet50



Hình 23: Ma trận nhầm lẫn (Confusion Matrix) của ResNet50

- Mô hình nhận diện đúng 353/390 ca viêm phổi (Recall = 90.5%).
- Có 25 ảnh phổi bình thường bị nhầm là viêm (False Positive) và 37 ảnh viêm bị nhầm là bình thường (False Negative).
- Tỷ lệ nhầm lẫn thấp hơn so với CNN, chứng tỏ ResNet50 tổng quát hoá tốt hơn trên tập test.

5.4. Mô hình Xception

5.4.1. Cấu trúc mô hình

Mô hình Xception được xây dựng dựa trên kiến trúc gốc của Xception (Extreme Inception). Một phiên bản mở rộng của Inception do

Model: "model_1"

Layer (type)	Output Shape	Param #
input_4 (InputLayer)	[(None, 224, 224, 3)]	0
xception (Functional)	(None, 7, 7, 2048)	20861480
global_average_pooling2d_1 (GlobalAveragePooling2D)	(None, 2048)	0
dropout_2 (Dropout)	(None, 2048)	0
dense_2 (Dense)	(None, 256)	524544
dropout_3 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 3)	771
=====		
Total params: 21,386,795		
Trainable params: 11,079,643		
Non-trainable params: 10,307,152		

Hình 24: Cấu trúc mô hình Xception

FrançoisChollet đề xuất, sử dụng hoàn toàn các lớp Depthwise Separable Convolution giúp giảm số tham số và tăng khả năng trích xuất đặc trưng.

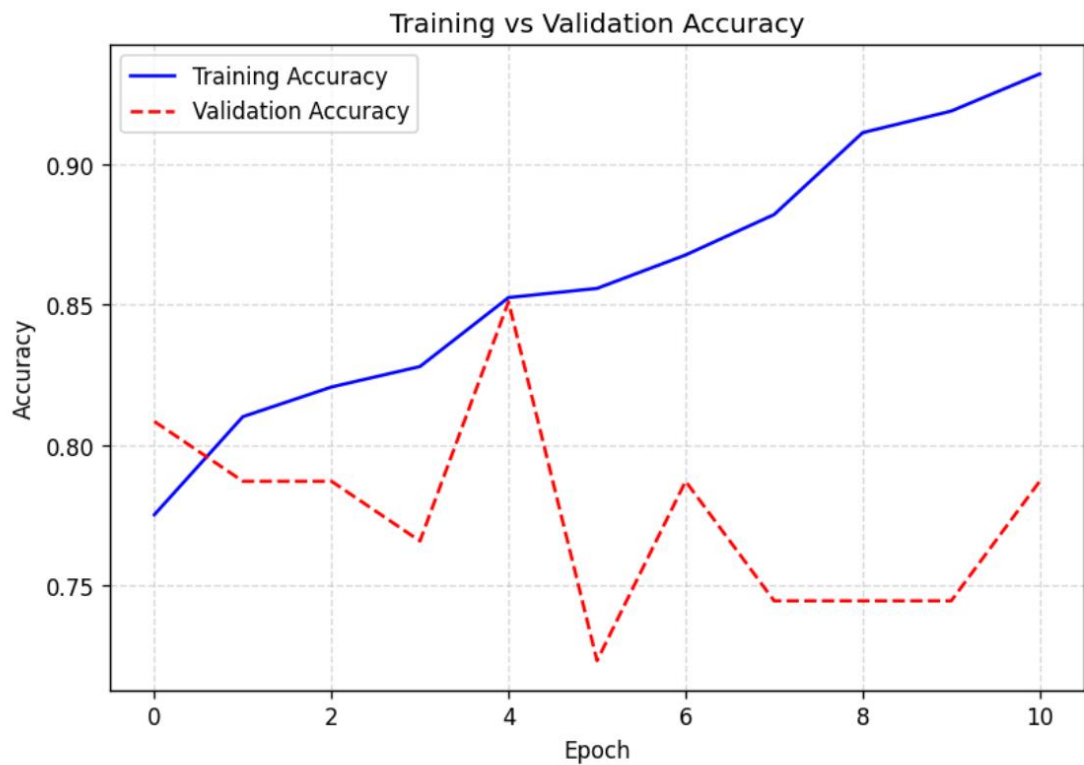
Tổng cộng 21,386,795 tham số, trong đó 11,079,643 tham số được huấn luyện và 10,307,152 tham số cố định (non-trainable).

Đây là mô hình fine-tuning kết hợp transfer learning, vừa giữ đặc trưng sâu từ ImageNet vừa điều chỉnh phù hợp với dữ liệu X-quang.

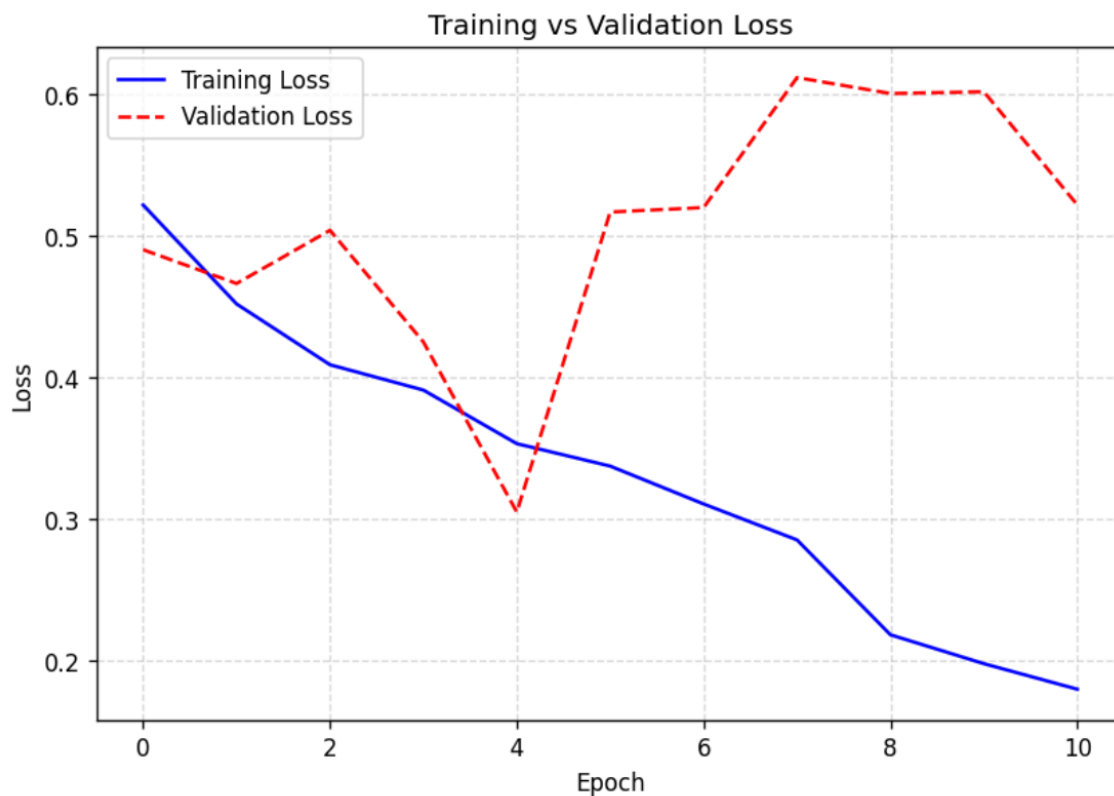
5.4.2. Kết quả huấn luyện của Xception

Mô hình được huấn luyện trong **10 epoch** với kích thước ảnh $224 \times 224 \times 3$. Biểu đồ dưới đây thể hiện tiến trình huấn luyện = Sau giai đoạn fine-tuning, mô hình hội tụ ổn định với:

- **Training Accuracy** tăng đều từ khoảng 77% lên 91%, cho thấy mô hình học ổn định.
- **Validation Accuracy** dao động trong khoảng **75–85%**, cho thấy mô hình có khả năng khái quát tương đối tốt.
- Training Loss giảm mạnh từ 0.52 xuống 0.18, trong khi Validation Loss dao động quanh 0.5–0.6, thể hiện mô hình chưa overfit rõ rệt nhưng vẫn còn nhiều nhẹ trong quá trình học.



Hình 25: Biểu đồ Training vs Validation Accuracy



Hình 26: Biểu đồ Training vs Validation Loss

5.4.3. Báo cáo phân loại

Đánh giá trên 624 ảnh (234 NORMAL, 242 BACTERIA, 148 VIRUS) cho kết quả:

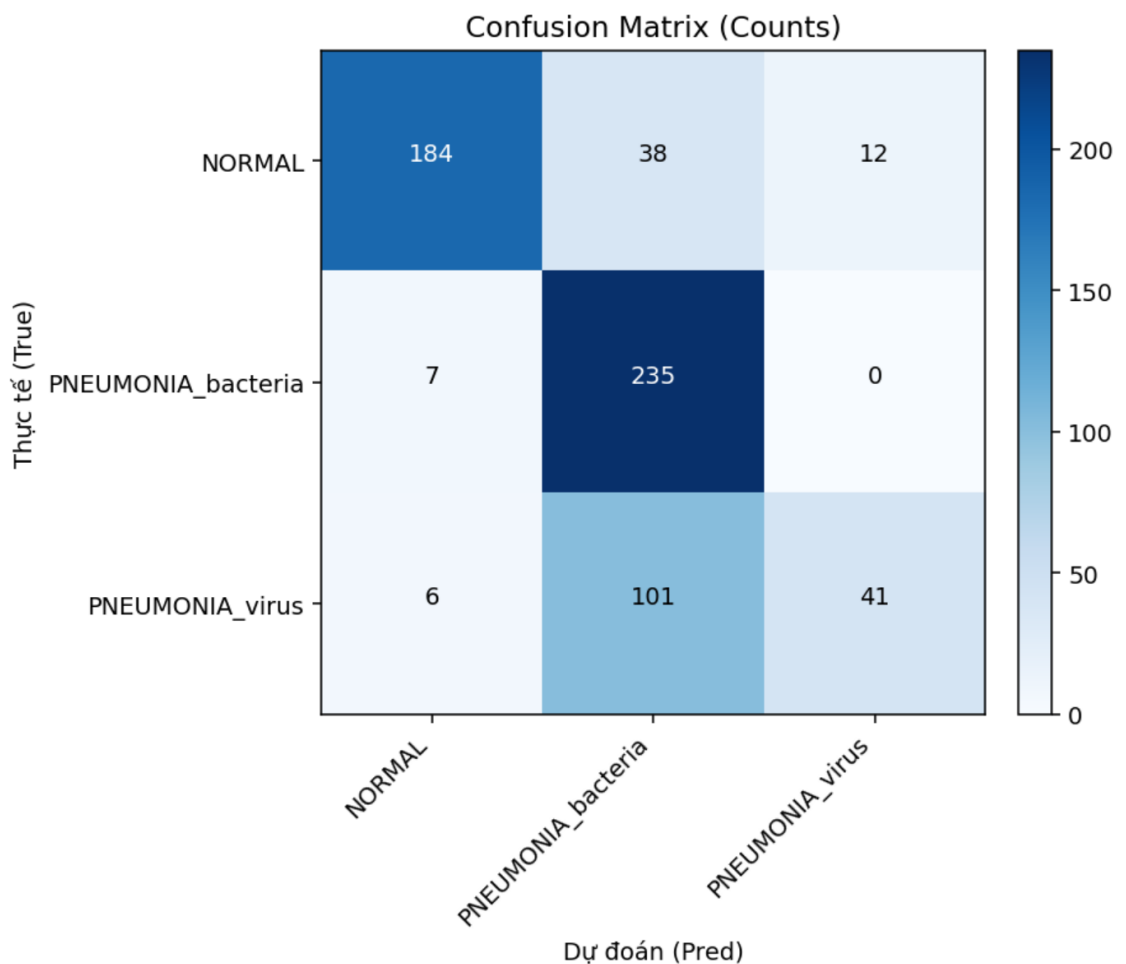
===== CLASSIFICATION REPORT =====				
	precision	recall	f1-score	support
NORMAL	0.9340	0.7863	0.8538	234
PNEUMONIA_bacteria	0.6283	0.9711	0.7630	242
PNEUMONIA_virus	0.7736	0.2770	0.4080	148
accuracy			0.7372	624
macro avg	0.7786	0.6781	0.6749	624
weighted avg	0.7774	0.7372	0.7128	624

Hình 27: Classification Report chi tiết của mô hình Xception

Mặc dù độ chính xác tổng thể ở mức trung bình, mô hình vẫn nhận diện rất tốt các ca viêm phổi do vi khuẩn với Recall = 0.9711, chứng tỏ khả năng phát hiện vùng tổn thương mạnh.

Tuy nhiên, lớp viêm do virus có Recall thấp (0.277), thể hiện mô hình chưa đủ khả năng phân biệt được đặc trưng giữa *bacteria* và *virus* do độ tương đồng cao giữa hai loại ảnh.

5.4.4. Ma trận nhầm lẫn (Confusion Matrix)



28: Ma trận nhầm lẫn ba lớp (Normal – Bacteria – Virus) của Xception

235 ảnh viêm do vi khuẩn được nhận diện đúng → rất mạnh ở nhóm này.

101 ảnh virus bị nhầm thành bacteria → mô hình thiếu khả năng tách biệt đặc trưng giữa hai loại viêm phổi.

38 ảnh bình thường bị nhầm là viêm, có thể do vùng phổi sáng hoặc noise cao.

5.4.5. Nhận xét tổng quan

Về OVERALL METRICS:

Accuracy : 0.7372
Macro-F1 : 0.6749
Weighted-F1: 0.7128

Hình 29: OVERALL METRICS của Xception

- Mô hình Xception đạt Accuracy = 73.7%, Weighted F1 = 0.7128, và cho thấy tiềm năng tốt trong phát hiện viêm phổi do vi khuẩn.
- Tuy nhiên, khả năng phân biệt giữa *virus* và *bacteria* còn hạn chế, cần điều chỉnh thêm về dữ liệu và kỹ thuật regularization.
- Dù chưa đạt hiệu năng cao nhất, Xception cho kết quả hiểu sâu đặc trưng hình ảnh vùng phổi, thể hiện đúng vai trò của mô hình kiến trúc sâu trong phân loại đa lớp.

5.5. Kết quả mô hình EfficientNetB0

5.5.1. Cấu trúc mô hình

Mô hình được xây dựng dựa trên kiến trúc EfficientNetB0, một mô hình tối ưu hóa giữa độ chính xác và số tham số thông qua kỹ thuật compound scaling (cân bằng chiều sâu – độ rộng – độ phân giải).

Cấu trúc chính gồm:

- Khởi tiên xử lý: Rescaling, Normalization.
- EfficientNetB0 backbone (ImageNet pre-trained) – học đặc trưng sâu của ảnh X-quang
- GlobalAveragePooling2D → Dense(3, Softmax) cho phân loại 3 lớp

Model: "model"

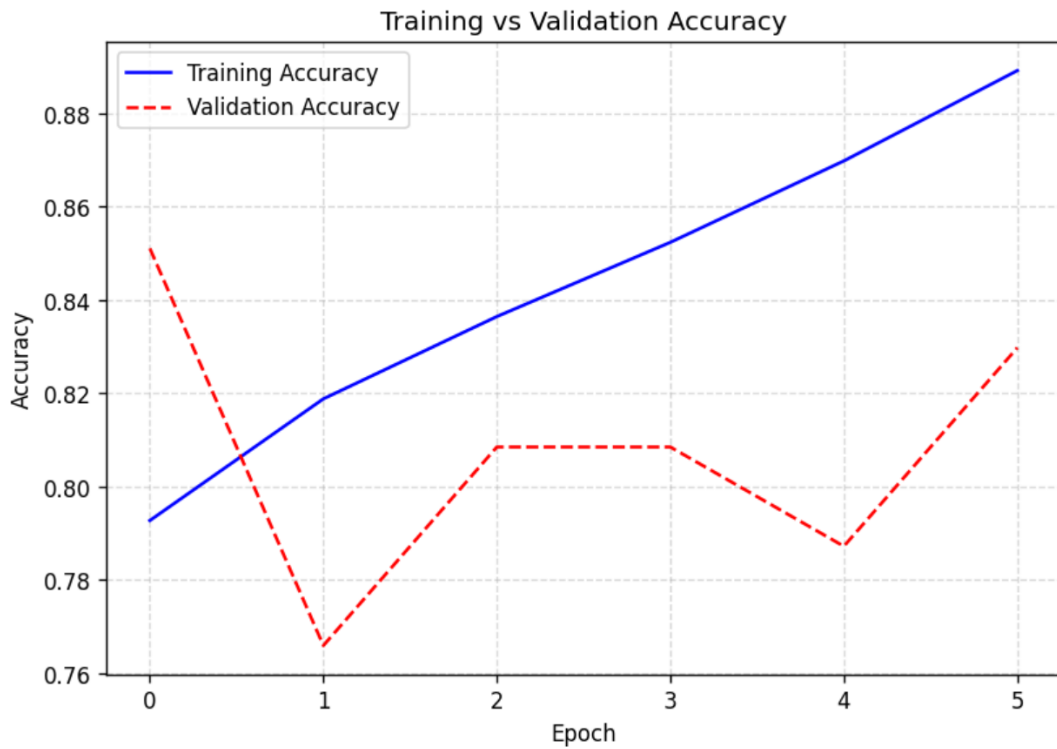
Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 224, 224, 3)]	0	[]
rescaling (Rescaling)	(None, 224, 224, 3)	0	['input_1[0][0]']
normalization (Normalization)	(None, 224, 224, 3)	7	['rescaling[0][0]']
rescaling_1 (Rescaling)	(None, 224, 224, 3)	0	['normalization[0][0]']
stem_conv_pad (ZeroPadding2D)	(None, 225, 225, 3)	0	['rescaling_1[0][0]']
stem_conv (Conv2D)	(None, 112, 112, 32)	864	['stem_conv_pad[0][0]']
stem_bn (BatchNormalization)	(None, 112, 112, 32)	128	['stem_conv[0][0]']
stem_activation (Activation)	(None, 112, 112, 32)	0	['stem_bn[0][0]']
block1a_dwconv (DepthwiseConv2D)	(None, 112, 112, 32)	288	['stem_activation[0][0]']
...			
Total params: 4053414 (15.46 MB)			
Trainable params: 3843 (15.01 KB)			
Non-trainable params: 4049571 (15.45 MB)			

Hình 30: Cấu trúc EfficientNetB0 backbone.

Tổng số tham số huấn luyện khoảng 5.3 triệu, nhỏ hơn nhiều so với ResNet50 hay Xception (~20–24 triệu), giúp tốc độ huấn luyện nhanh hơn và phù hợp triển khai thực tế.

5.5.2. Kết quả huấn luyện mô hình EfficientNetB0

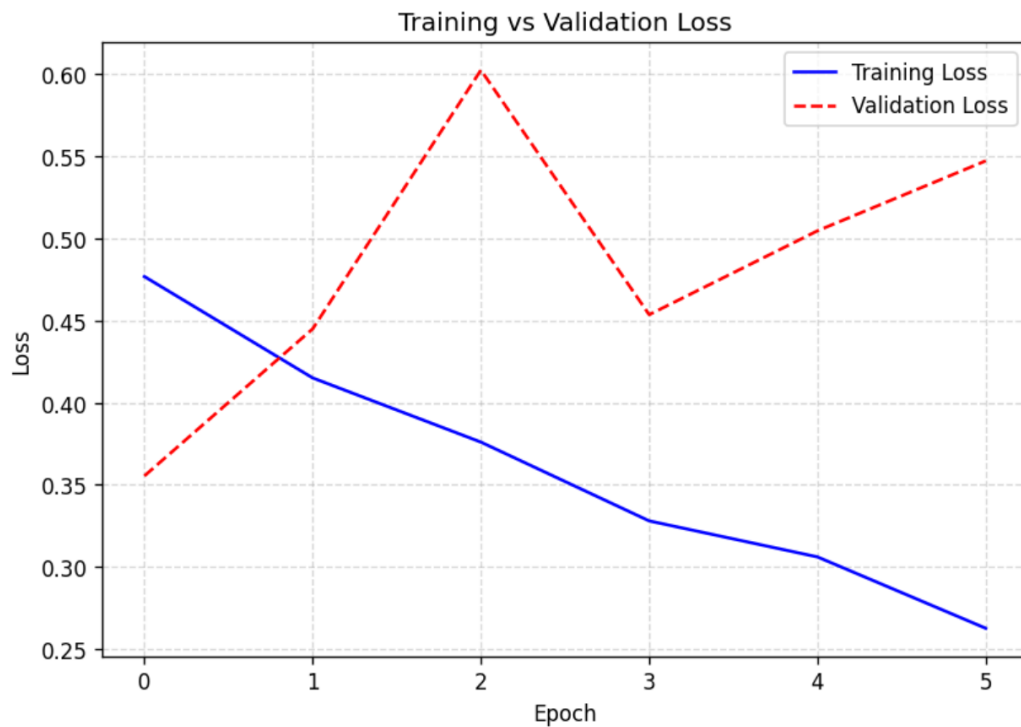
Mô hình được huấn luyện trên dữ liệu ảnh 224×224×3 trong 20 epoch.



Hình 31: Biểu đồ Training vs Validation Accuracy

Độ chính xác huấn luyện (Training Accuracy) tăng dần từ khoảng 79% lên gần 89% sau 5 epoch.

Độ chính xác kiểm định (Validation Accuracy) dao động quanh 77%–85%, thể hiện sự ổn định tương đối và không có dấu hiệu overfitting nghiêm trọng.



Hình 32: Biểu đồ Training vs Validation Loss

Biểu đồ Loss cho thấy Training Loss giảm đều từ ~0.48 xuống ~0.27, trong khi Validation Loss dao động trong khoảng 0.35–0.6, với xu hướng chung là ổn định dần.

5.5.3. Kết quả đánh giá trên tập kiểm thử

- OVERALL METRICS:

Accuracy : 0.8125
Macro-F1 : 0.8007
Weighted-F1: 0.8102

Hình 33: OVERALL METRICS của EfficientNetB0

So với Xception (Accuracy = 73.7%), mô hình EfficientNetB0 đã cải thiện đáng kể khả năng tổng quát, thể hiện độ ổn định cao hơn trên cả ba lớp.

5.5.4. Báo cáo phân loại

	precision	recall	f1-score	support
NORMAL	0.9353	0.6795	0.7871	234
PNEUMONIA_bacteria	0.8105	0.9545	0.8767	242
PNEUMONIA_virus	0.6923	0.7905	0.7382	148
accuracy			0.8125	624
macro avg	0.8127	0.8082	0.8007	624
weighted avg	0.8293	0.8125	0.8102	624

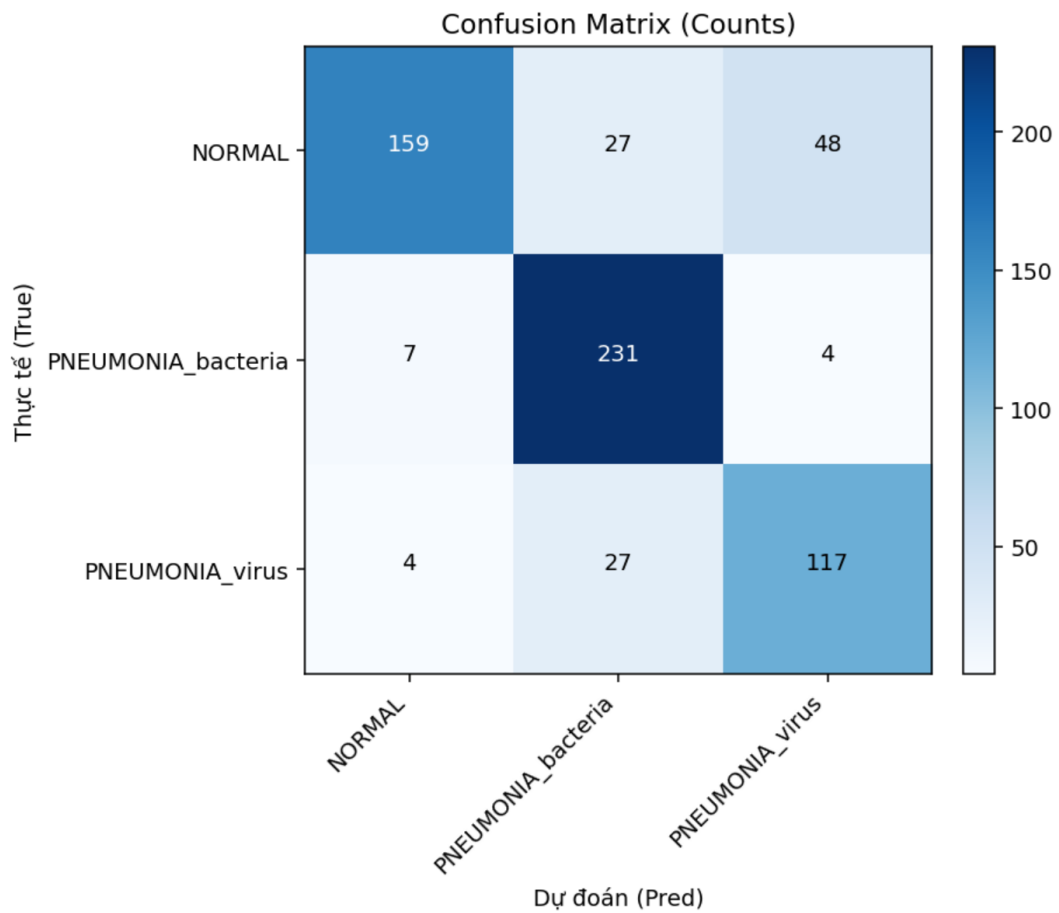
Hình 34: Classification Report chi tiết của EfficientNetB0

- Lớp NORMAL: Precision rất cao (0.93) → mô hình dự đoán "bình thường" rất chắc chắn, tuy nhiên Recall = 0.68 cho thấy vẫn bỏ sót một số ca bình thường.
- Lớp BACTERIA: Recall = 0.95 – gần như phát hiện đầy đủ các ca viêm do vi khuẩn, tương tự Xception nhưng ổn định hơn.
- Lớp VIRUS: F1 = 0.74 – cao hơn đáng kể so với Xception (0.40), chứng tỏ EfficientNet đã cải thiện khả năng phân biệt giữa virus và bacteria.

EfficientNetB0 đạt sự cân bằng tốt nhất giữa độ chính xác các lớp, giảm tình trạng nhầm giữa bacteria và virus.

5.5.5. Ma trận nhầm lẫn (Confusion Matrix)

- 231/242 ảnh bacteria được nhận diện đúng → Recall rất cao (0.9545).
- Lớp virus cải thiện rõ: 117/148 ảnh đúng (so với chỉ 41 ảnh đúng ở Xception).
- Một phần ảnh bình thường (48/234) vẫn bị nhầm sang virus, có thể do các vùng phổi có cấu trúc mờ tương tự bệnh nhẹ.



Hình 35; Ma trận nhầm lẫn ba lớp (Normal – Bacteria – Virus) của EfficientNetB0

5.6. So sánh và kết luận tổng hợp các mô hình

Để đánh giá toàn diện hiệu quả của các mô hình học sâu trong bài toán phân loại ảnh X-quang phổi, nhóm đã tiến hành huấn luyện và kiểm thử bốn mô hình khác nhau: CNN cơ bản, ResNet50, Xception và EfficientNetB0.

Mô hình	Số lớp phân loại	Accuracy	Macro-F1	Weighted-F1
CNN cơ bản	2	90.22	0.8907	0.8996
ResNet50	2	90.06	0.8951	0.9011
Xception	3	73.72	0.6749	0.7128

EfficientNetB0	3	81.25	0.8007	0.8102
----------------	---	-------	--------	--------

Kết luận chung:

- Các mô hình học sâu đều cho kết quả khả quan trong việc nhận dạng ảnh X-quang phổi, với độ chính xác tổng thể đạt từ 73% đến 90%.
- Với bài toán nhị phân, cả CNN cơ bản và ResNet50 đạt độ chính xác cao và ổn định; trong đó ResNet50 có khả năng khái quát tốt hơn nhờ kiến trúc sâu hơn
- Với bài toán đa lớp, EfficientNetB0 vượt trội hơn Xception, đạt Accuracy 81.25% và Weighted-F1 0.81, thể hiện sự cân bằng giữa hiệu năng và tốc độ.
- Xception tuy độ chính xác thấp hơn nhưng vẫn hữu ích khi cần trích xuất vùng ảnh đặc trưng phục vụ Grad-CAM hoặc hỗ trợ bác sĩ giải thích kết quả.

Tổng kết:

- Trong bốn mô hình được thử nghiệm, EfficientNetB0 được đánh giá là tối ưu nhất cho bài toán nhận diện viêm phổi đa lớp trên ảnh X-quang, nhờ hiệu năng cao, khả năng tổng quát tốt, và cấu trúc gọn nhẹ phù hợp cho triển khai thực tế.

5.7. Triển khai dự án với model EfficientNetB0

5.7.1. Kiến trúc tổng thể

- Frontend (Vite + React): trang “Upload ảnh → Phân tích → Kết quả”. FE gọi API BE, nhận JSON + ảnh base64, hiển thị xác suất từng lớp, nhãn top-1, ảnh gốc và Grad-CAM.
- Backend (FastAPI + TensorFlow/Keras): nạp model EfficientNetB0, xử lý ảnh (resize 224×224, preprocess), suy luận, tạo Grad-CAM, đóng gói JSON trả về.

5.7.2 Vai trò Backend (FastAPI)

- Khởi động dịch vụ
- Tiền xử lý ảnh
- Suy luận & hậu xử lý
- Giao ước dữ liệu (response)
- Bảo mật & độ tin cậy (tối thiểu)

5.7.3 Vai trò Frontend (Vite + React)

- Người dùng kéo-thả/chọn file để hiển thị preview.
- Trải nghiệm & giải thích:
 - o Nhấn mạnh độ tin cậy (top-1) + thanh progress cho từng lớp
 - o Kèm cảnh báo/khuyến nghị: AI chỉ hỗ trợ tham khảo, không thay thế bác sĩ.
- Cấu hình & môi trường
 - o FE đọc VITE_API_BASE để biết URL BE

5.7.4 Triển khai & vận hành

Môi trường: Python 3.10/TF 2.10, FastAPI + Uvicorn/Gunicorn.

Hiệu năng:

- o EfficientNetB0 nhẹ, thời gian suy luận $< \sim 1s$ /ảnh (CPU tốt); có thể bật batching hoặc cache nếu cần.
- o Tạo Grad-CAM tốn thêm vài trăm ms—chấp nhận được cho tác vụ đơn chiếc.

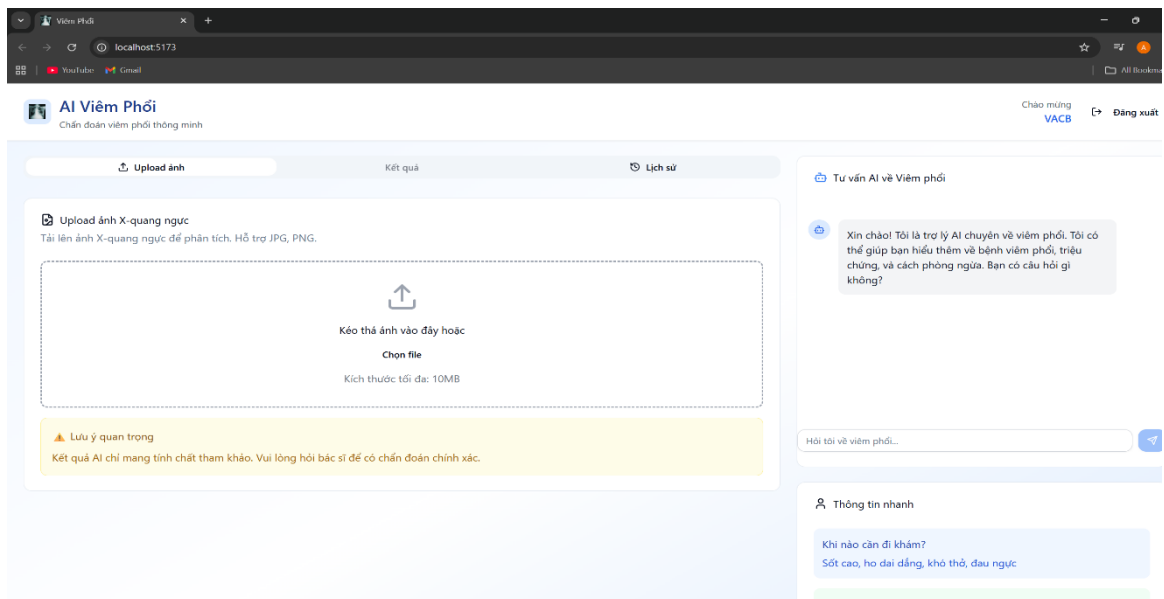
5.7.5 Lý do chọn EfficientNetB0 để triển khai

Cân bằng tốt giữa độ chính xác (Accuracy 81.25%, Weighted-F1 0.81) và tốc độ/nhẹ. Phù hợp sản phẩm web.

Hiệu năng đa lớp: phân biệt virus tốt hơn Xception.

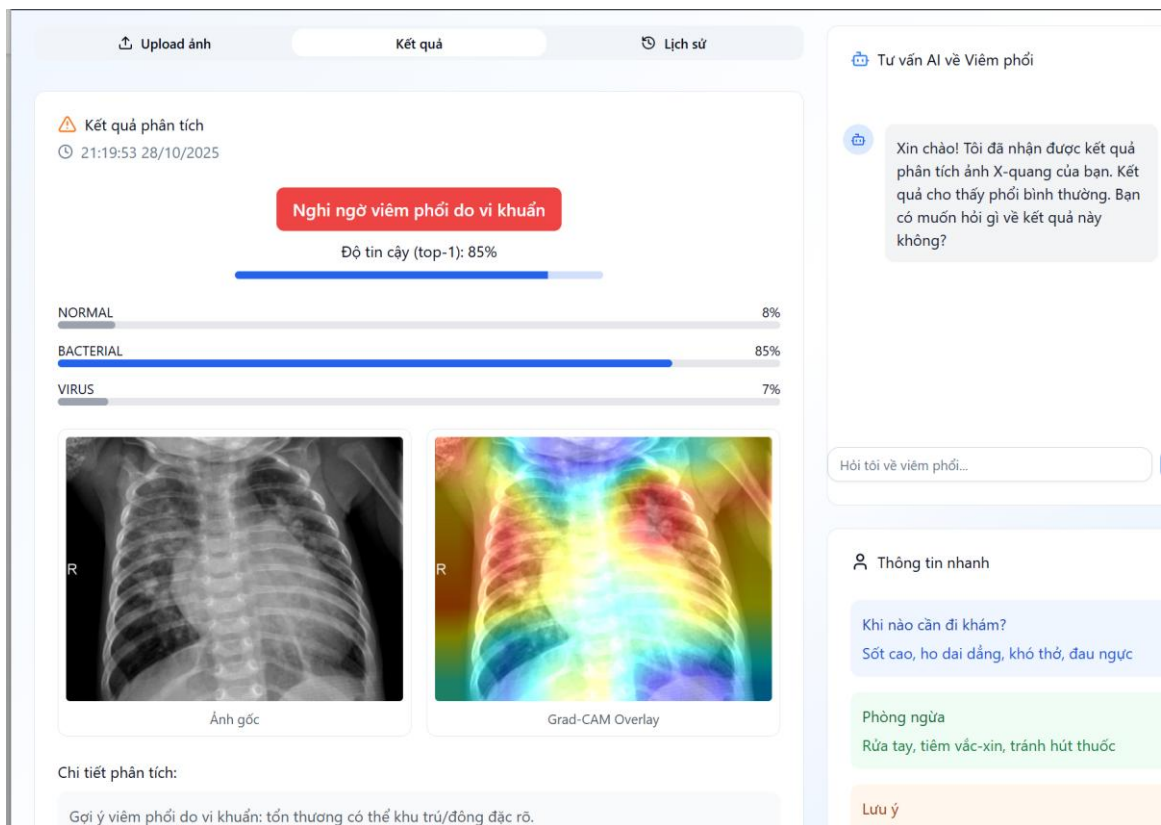
5.7.6. Giao diện hệ thống và minh họa kết quả triển khai:

- Trang Upload ảnh – người dùng chọn hoặc kéo thả ảnh X-ray.



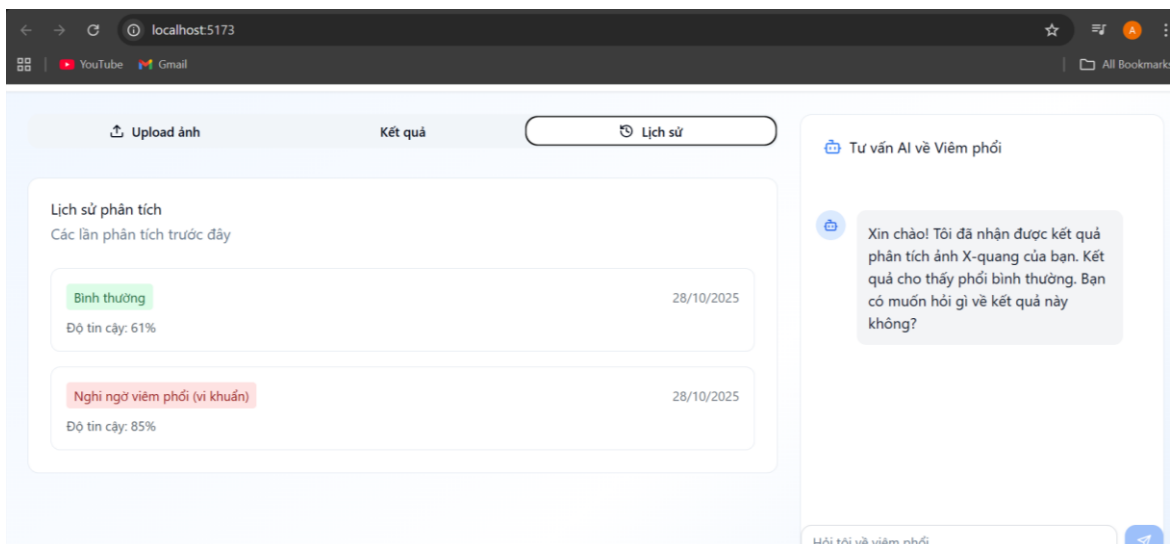
Hình 36: Giao diện frontend Upload Ảnh

- Trang Kết quả hiển thị nhãn chẩn đoán, độ tin cậy (%).
 - o Hiển thị kết quả rõ ràng, có minh họa Grad-CAM.
 - o Cung cấp trải nghiệm thân thiện, có hướng dẫn và cảnh báo an toàn.



Hình 37: Trang Fontend hiển thị kết quả

- Trang lịch sử phân tích: hiển thị thời gian và lưu lại lịch sử người dùng



Hình 38: Trang Fontend hiển thị lịch sử sử dụng

CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1. Kết luận

Đề tài “Khai phá dữ liệu ảnh X-quang phổi (Chest -Xray) để phát hiện viêm phổi bằng Deep_Learning” đã hoàn thành các mục tiêu đặt ra, từ khâu tiền xử lý dữ liệu, huấn luyện mô hình, đến triển khai ứng dụng thực tế trên nền web.

Đã xây dựng và so sánh bốn mô hình học sâu gồm CNN cơ bản, ResNet50, Xception, và EfficientNetB0.

EfficientNetB0 thể hiện hiệu năng tối ưu nhất trong bài toán đa lớp (Normal – Bacteria – Virus) với Accuracy 81.25%, Weighted-F1 0.81, và thời gian suy luận nhanh, phù hợp cho triển khai thực tế.

Đã tích hợp mô hình EfficientNetB0 vào hệ thống web.

Hệ thống web hoạt động ổn định, cho phép người dùng tải ảnh X-quang, nhận kết quả phân tích, xem Grad-CAM minh họa vùng tổn thương và nhận tư vấn hướng dẫn y tế cơ bản.

Kết quả cho thấy tính khả thi của việc ứng dụng học sâu vào hỗ trợ chẩn đoán y khoa, góp phần giảm tải cho bác sĩ và nâng cao khả năng sàng lọc sớm bệnh viêm phổi.

6.2. Hạn chế

Mặc dù hệ thống đạt được những kết quả tích cực, đề tài vẫn còn một số hạn chế:

- Tập dữ liệu chưa thực sự cân bằng (số lượng ảnh PNEUMONIA_virus còn ít), ảnh hưởng đến khả năng phân biệt giữa hai loại viêm.
- Chưa thực hiện được giai đoạn fine-tuning nâng cao hoặc data augmentation
- chuyên sâu để tăng độ khái quát của mô hình.

6.3. Hướng phát triển

Trong tương lai, nhóm dự kiến mở rộng và hoàn thiện hệ thống theo các hướng:

- Cải thiện mô hình học sâu:
 - o Mở rộng tập dữ liệu (đặc biệt là ảnh *virus*).
 - o Thử nghiệm các kiến trúc tiên tiến hơn như EfficientNetV2, EfficientNetB5,...
- Tối ưu và mở rộng hệ thống:
 - o Cải thiện thời gian suy luận và giao diện người dùng, bổ sung thống kê, báo cáo, hoặc lưu trữ hồ sơ điện tử.
 - o Tích hợp AI trợ lý y tế: Mở rộng chatbot tư vấn.

TÀI LIỆU THAM KHẢO

<https://viso.ai/deep-learning/xception-model/> by [Gaudenz Boesch](#)

<https://www.mdpi.com/2075-4418/13/2/248>

<https://viso.ai/deep-learning/efficientnet/>

<https://www.mdpi.com/2504-2289/5/4/73>

<https://www.kaggle.com/code/ichigoe/efficientnet-with-grad-cam-visualization>

<https://www.sciencedirect.com/science/article/abs/pii/S0168169920300442>

<https://viblo.asia/p/efficientnet-cach-tiep-can-moi-ve-model-scaling-cho-convolutional-neural-networks-Qbq5QQzm5D8>

<https://viblo.asia/p/paper-reading-xception-phien-ban-nang-cap-cua-inception-v3-MkNLr1ZoJgA>

<https://www.ultralytics.com/vi/blog/what-is-resnet-50-and-what-is-its-relevance-in-computer-vision>