

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

NGUYỄN KIỀU PHÁT

**MÔ HÌNH HỌC SÂU PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRÊN  
ẢNH Y KHOA HỖ TRỢ TRONG CHẨN ĐOÁN UNG THƯ**

**LUẬN VĂN THẠC SĨ: NGÀNH KHOA HỌC MÁY TÍNH**

Tp. Hồ Chí Minh, năm 2018

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

**NGUYỄN KIỀU PHÁT**

**MÔ HÌNH HỌC SÂU PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRÊN  
ẢNH Y KHOA HỖ TRỢ TRONG CHẨN ĐOÁN UNG THƯ**

Chuyên ngành: Khoa học máy tính  
Mã số chuyên ngành: 60480101

**LUẬN VĂN THẠC SĨ: NGÀNH KHOA HỌC MÁY TÍNH**

**NGƯỜI HƯỚNG DẪN KHOA HỌC  
PGS.TS LÊ HOÀNG THÁI**

Tp. Hồ Chí Minh, năm 2018

## **LỜI CAM ĐOAN**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Các số liệu, kết quả nêu trong Luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tôi xin cam đoan rằng mọi sự giúp đỡ cho việc thực hiện Luận văn này đã được cảm ơn và các thông tin trích dẫn trong Luận văn đã được chỉ rõ nguồn gốc.

Tôi xin cam đoan nội dung và kết quả của luận văn này xuất phát từ công sức và quá trình lao động, nghiên cứu.

**HỌC VIÊN THỰC HIỆN**

**Nguyễn Kiều Phát**

## MỤC LỤC

<b>DANH MỤC CÁC TỪ VIẾT TẮT .....</b>	<b>4</b>
<b>DANH MỤC HÌNH ẢNH .....</b>	<b>5</b>
<b>DANH MỤC BẢNG BIỂU.....</b>	<b>8</b>
<b>MỞ ĐẦU.....</b>	<b>9</b>
1. Lý do chọn đề tài .....	9
2. Mục đích nghiên cứu .....	10
3. Mục tiêu nghiên cứu .....	11
4. Đối tượng nghiên cứu và phạm vi nghiên cứu .....	11
5. Phương pháp nghiên cứu .....	11
6. Đóng góp của đề tài .....	11
7. Nội dung luận văn.....	12
<b>CHƯƠNG 1 - TỔNG QUAN VỀ BÀI TOÁN PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRONG ẢNH Y KHOA .....</b>	<b>13</b>
1.1. Tổng quan về bài toán phân lớp.....	13
1.1.1. Khái niệm về phân lớp .....	13
1.1.2. Bài toán phân loại ảnh.....	13
1.1.3. Bài toán phát hiện bất thường trên hình ảnh y khoa .....	14
1.2. Tình hình nghiên cứu trong nước .....	15
1.3. Tình hình nghiên cứu ngoài nước .....	16
<b>CHƯƠNG 2 - CƠ SỞ LÝ THUYẾT.....</b>	<b>18</b>
2.1. Mạng mạng nơron nhân tạo .....	18
2.1.1. Các thành phần cơ bản của mạng nơron nhân tạo .....	18
2.1.2. Cấu trúc mạng nơron nhân tạo – ANN.....	21
2.1.3. Huấn luyện mạng ANN .....	22
2.1.4. Hàm mục tiêu.....	24
2.1.5. Mạng truyền thẳng và thuật toán lan truyền ngược.....	25
2.2. Kỹ thuật phân lớp ảnh sử dụng Convolutional Neural Network.....	29

2.3. Hàm Softmax .....	48
2.4. Vấn đề quá khớp (over-fitting).....	49
2.5. Đánh giá mô hình phân lớp .....	50
2.6. Ảnh y khoa.....	54
2.6.1. Giới thiệu: .....	54
2.6.2. Hình ảnh X-Quang.....	55
2.6.2.1. Sự ghi hình X-Quang.....	55
<b>CHƯƠNG 3 - XÂY DỰNG MÔ HÌNH HỌC SÂU PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRÊN HÌNH ẢNH X- QUANG PHỔI.....</b>	<b>57</b>
3.1. Kiến trúc mô hình Multi-CNNs.....	57
3.1.1. Giới thiệu .....	57
3.1.2. Các thành phần CNN của mô hình đề xuất .....	58
3.2. Các quy tắc kết hợp (fusion rules) dựa trên cây kết luận .....	67
<b>CHƯƠNG 4 - THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ .....</b>	<b>70</b>
4.1. Dữ liệu thực nghiệm .....	70
4.1.1. Bộ cơ sở dữ liệu hình ảnh x-quang ngực.....	70
4.1.2. Tiền xử lý hình ảnh và chuẩn hóa dữ liệu huấn luyện và kiểm tra...71	71
4.2. Quá trình thực nghiệm đánh giá mô hình đề xuất .....	72
4.2.1. Môi trường thực nghiệm.....	72
4.2.2. Quá trình huấn luyện các CNN thành phần của mô hình .....	73
4.3. Kết quả thực nghiệm mô hình đề xuất.....	75
<b>CHƯƠNG 5 - KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....</b>	<b>78</b>
5.1. Kết luận.....	83
5.2. Hướng phát triển.....	83
<b>DANH MỤC CÔNG TRÌNH CỦA TÁC GIẢ .....</b>	<b>85</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>86</b>
<b>PHỤ LỤC .....</b>	<b>89</b>

## DANH MỤC CÁC TỪ VIẾT TẮT

KÝ HIỆU	TÊN TIẾNG ANH
ANN	Artificial Neural Network
AB-Chest X-rays	An Bình hospital - Chest X-ray Database
CNN	Convolution Neural Network
DNN	Deep Neural Networks
FC	Fully Connected
FN	False Negative
FP	False Positive
FR1	Quy tắc kết hợp dựa trên việc tính trung bình giá trị xác xuất.
FR2	Quy tắc kết hợp dựa trên tổng thể.
FR3	Quy tắc kết hợp dựa trên các vùng xung đột.
Multi-CNNs	Multiple Convolutional Neural Network
MLP	Multilayer Perceptron Network
NPV	Khả năng phân loại lớp Negative chính xác của mô hình
PPV	Khả năng phân loại lớp Positive chính xác của mô hình
RELU	Rectified Linear Unit
RGB	Red Green Blue
SVM	Support Vector Machines
SZ-Chest X-rays	The Shenzhen set - Chest X-ray Database
TN	True Negative
TNR	Tỷ lệ nhận diện Negative đúng trong tổng số các trường hợp Negative
TP	True Positive
TPR	Tỷ lệ nhận diện Positive đúng trên tổng số các trường hợp Positive

## **DANH MỤC HÌNH ẢNH**

- Hình 1-1: Mô hình bài toán phân loại hình ảnh
- Hình 1-2: Mô hình tổng quát bài toán phát hiện bất thường hình ảnh y khoa
- Hình 2-1: Mô hình mạng nơron sinh học
- Hình 2-2: Cấu trúc 1 nơron nhân tạo
- Hình 2-3: Một số sơ đồ liên kết của mạng nơron
- Hình 2-4: Mô hình học có giám sát
- Hình 2-5: Mạng nơron truyền thẳng nhiều lớp
- Hình 2-6: Ảnh hưởng của kích thước mẫu
- Hình 2-7: Mô hình hàm softmax cho mạng Neural
- Hình 2-8: Cấu trúc cơ bản của mô hình CNN
- Hình 2-9: Kiến trúc của CNN. Nguồn: [21]
- Hình 2-10: Biểu diễn hình ảnh đầu vào và bộ lọc dưới dạng ma trận
- Hình 2-11: Tính tích chập của ảnh đầu vào với bộ lọc  $3 \times 3$ .
- Hình 2-13: Kết quả sau khi thực xong phép toán tích chập
- Hình 2-14: Thực hiện phép tích chập với một bộ lọc duy nhất
- Hình 2-15: Mô tả hai bản đồ tính năng tương ứng với hai bộ lọc
- Hình 2-16: Mô tả tính tích chập 3 chiều
- Hình 2-17: Mô tả hoạt động tại tầng convolution trong mô hình CNN với bộ lọc kích thước  $5 \times 5$
- Hình 2-18: Tích chập với bước trượt = 2 (stride = 2)
- Hình 2-19: Tích chập với bước trượt = 1 (stride = 2) và Padding = 1
- Hình 2-20: Biểu diễn của công thức 2.13 trên hệ trực tọa độ
- Hình 2-21: Biểu diễn của công thức 2.14 trên hệ trực tọa độ
- Hình 2-22: Minh họa về tầng pool và downsampling
- Hình 2-23: Minh họa tầng pool sử dụng hàm ReLU, phương pháp Max pooling và bộ lọc kích thước  $2 \times 2$
- Hình 2-24: Mô tả các giá trị trong mô hình CNN sử dụng để xác định hàm softmax

Hình 2-25: Minh họa tại tầng softmax của mô hình CNN với bộ dữ liệu MNIST

Hình 2-26: Gradient gốc với tỷ lệ học tập nhỏ (trên) và lớn (dưới). Nguồn Andrew Ng's Machine Learning course on Coursera [27]

Hình 2-27: Ảnh hưởng của giá trị learning rate đến độ hội tụ của mô hình mạng [27]

Hình 2-28: Một số hình ảnh y khoa [26]

Hình 3-1: Mô tả mô hình học sâu Multi-CNNs

Hình 3-2: Kiến trúc mạng của thành phần CNN-128F của mô hình Multi-CNNs

Hình 3-3: Mô tả hình ảnh đầu vào được đưa vào và tính toán ở tầng Input của kiến trúc CNN-128F

Hình 3-4: Mô tả trực quan tầng conv của kiến trúc mạng CNN-128F

Hình 3-5: Mô tả trực quan ReLU của kiến trúc mạng CNN-128F

Hình 3-6: Mô tả trực quan tầng Pool của kiến trúc mạng CNN-128F

Hình 3-7: Mô tả trực quan tầng conv của layer 2 của kiến trúc mạng CNN-128F

Hình 3-8: Mô tả trực quan tầng ReLU của layer 2 của kiến trúc mạng CNN-128F

Hình 3-9: Mô tả trực quan tầng Pool của layer 2 của kiến trúc mạng CNN-128F

Hình 3-10: Mô tả trực quan tầng conv của layer 3 của kiến trúc mạng CNN-128F

Hình 3-11: Mô tả trực quan tầng ReLU và Pool của layer 3 của kiến trúc mạng CNN-128F

Hình 3-12: Mô tả trực quan lớp FC của kiến trúc mạng CNN-128F

Hình 3-13: Mô tả trực các tính toán của hàm softmax của kiến trúc mạng CNN-128F

Hình 3-14: Kiến trúc mạng của thành phần CNN-64L và CNN-64R của mô hình Multi-CNNs

Hình 3-17: Mô tả trực quan tầng conv của kiến trúc mạng CNN-64L/R

Hình 3-18: Mô tả trực quan tầng ReLU của kiến trúc mạng CNN-64L/R

Hình 3-19: Mô tả trực quan tầng Pool của kiến trúc mạng CNN-64L/R

Hình 4-1: Các bước thực hiện tiền xử lý ảnh đầu vào

Hình 4-2: Mô tả quá trình ảnh vào batch dữ liệu để đưa vào hệ thống CNN

Hình 4-3: Biểu đồ thể hiện Độ chính xác của 3 thành phần CNN của mô hình ứng với bộ dữ liệu huấn luyện

Hình 4-4: Biểu đồ so sánh thời gian huấn luyện để đạt được độ chính xác cao nhất của mỗi mạng CNN thành phần.

Hình 4-5: Biểu đồ so sánh độ chính xác của mô hình trên từng tập dữ liệu ứng với từng phương pháp tính toán kết hợp.

Hình 4-6: Biểu đồ so sánh thời gian trung bình phân lớp 1 ảnh đầu vào của các CNN thành phần và mô hình Multi-CNNs

Hình 4-7: Biểu đồ thể hiện độ chính xác tổng quát của mô hình và độ chính xác trên từng lớp của mô hình trên tập dữ liệu AB-Chest X-rays khi thống kê kết quả phân lớp của mô hình với quy tắc kết hợp FR3.

Hình 4-8: Biểu đồ thể hiện độ chính xác tổng quát của mô hình và độ chính xác trên từng lớp của mô hình trên tập dữ liệu SZ-Chest X-rays khi thống kê kết quả phân lớp của mô hình với quy tắc kết hợp FR3.

## **DANH MỤC BẢNG BIỂU**

Bảng 2-1: Các hàm truyền cơ bản

Bảng 2-2: Mẫu kết quả tại tầng fc với bộ dữ liệu MNIST

Bảng 2-3: Định nghĩa True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) dựa trên confusion matrix chưa chuẩn hoá

Bảng 4-1: Mô tả chi tiết số lượng hình ảnh của từng bộ ảnh

Bảng 4-2: Mô tả chi tiết số lượng hình ảnh lựa chọn trong bộ dữ liệu để huấn luyện và kiểm tra cho từng thành phần của CNN của mô hình

Bảng 4-3: Kết quả huấn luyện các CNN thành phần của mô hình Mulit-CNNs

Bảng 4-4: Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu AB-Chest X-rays tương ứng với 3 mạng CNN thành phần

Bảng 4-5: Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu AB-Chest X-rays tương ứng với 3 luật kết hợp

Bảng 4-6: Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu SZ-Chest X-rays tương ứng với 3 mạng CNN thành phần.

Bảng 4-7: Kết quả kiểm tra tên trên 150 hình ảnh thuộc bộ dữ liệu SZ-Chest X-rays tương ứng với 3 luật kết hợp

Bảng 4-8: Thời gian trung bình của quá trình tính toán để đưa ra kết luận cuối cùng cho 1 ảnh đầu vào

Bảng 4-9: Biểu diễn các giá trị TP, FP, TN, FN trên tập dữ liệu AB-Chest X-rays

Bảng 4-10: Biểu diễn các giá trị TPR, FNR, FPR, TNR trên tập dữ liệu AB-Chest X-rays

Bảng 4-11: Biểu diễn các giá trị TP, FP, TN, FN trên tập dữ liệu SZ-Chest X-rays

Bảng 4-12: Biểu diễn các giá trị TPR, FNR, FPR, TNR trên tập dữ liệu SZ-Chest X-rays

# MỞ ĐẦU

## 1. Lý do chọn đề tài

Ngày nay, cùng với sự phát triển vượt bậc của lĩnh vực công nghệ thông tin thế giới kỹ thuật trong những năm gần đây, khối lượng dữ liệu và thông tin phát triển không ngừng đặc biệt là số lượng hình ảnh trong các lĩnh vực ngày càng lớn. Với một lượng hình ảnh ngày càng lớn như vậy, việc phân loại và phát hiện những vấn đề thông qua hình ảnh dần trở thành một nhu cầu tất yếu nhằm phục vụ cho công tác nghiên cứu phát triển các công cụ hỗ trợ con người giải quyết các vấn đề trong cuộc sống. Sự phát triển vượt bậc của các thiết bị đồ họa cũng như thiết bị lưu trữ cũng chính là một nhân tố tích cực thúc đẩy nghiên cứu các ứng dụng thực tế từ công nghệ xử lý ảnh số.

Những năm gần đây, ta đã chứng kiến được nhiều thành tựu vượt bậc trong ngành Thị giác máy tính (Computer Vision). Các hệ thống xử lý ảnh lớn như Facebook, Google, ... đã đưa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động.

Tự động phát hiện các đối tượng trong hình ảnh số là nhu cầu cấp thiết trong nhiều lĩnh vực khác nhau như: an ninh, nhận dạng đối tượng, kiểm soát giao thông và đặc biệt là phát hiện sớm những dấu hiệu bất thường dựa trên hình ảnh y khoa.

Hình ảnh y khoa sử dụng chủ yếu là các hình ảnh được chụp X-quang, chụp CT-cắt lớp, chụp MRI cộng hưởng từ và hình ảnh siêu âm nhằm cung cấp những hình ảnh quan trọng bên trong cấu trúc con người mà mắt thường không thể nhìn thấy được qua đó giúp các bác sĩ đưa ra những quyết định tốt nhất trong điều trị bệnh. Chẩn đoán hình ảnh đã và đang đóng vai trò quan trọng trong y khoa, các phương tiện hiện đại ngày nay cho phép chúng ta phát hiện sớm và chính xác hơn các bệnh lý, chúng cũng cho phép hạn chế các thủ thuật điều trị không cần thiết. Bằng cách đó, chẩn đoán hình ảnh đã tham gia ở tuyến đầu trong tiến trình chẩn đoán và điều trị bệnh của các bác sĩ. Hiện nay, hầu hết các bệnh viện, cơ sở khám chữa bệnh đều có

khoa chẩn đoán hình ảnh cùng các thiết bị máy móc hiện đại hỗ trợ và đương nhiên y học không thể thiếu chuyên ngành chẩn đoán hình ảnh.

Bài toán phát hiện những đối tượng bất thường trong hình ảnh y khoa là bài toán có ý nghĩa thực tiễn lớn nhằm hỗ trợ cho các bác sĩ trong việc chẩn đoán bệnh trong y học, kết quả phát hiện càng chính xác thì càng hiệu quả trong tiến trình chẩn đoán và điều trị của bác sĩ. Để có thể ứng dụng các phương pháp máy học để giải quyết bài toán này, chúng ta cần có những phương pháp và mô hình có độ chính xác cao. Có thể kể đến các phương pháp thông minh: K-Mean, kNN, AdaBoost, SVM, Mạng Neural, ... **Convolutional Neural Network** (CNN - Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay. Do đó, trong luận văn này, học viên tập trung nghiên cứu và đề xuất “*Mô hình học sâu phát hiện bất thường đậm độ trên hình ảnh X-Quang hỗ trợ trong chẩn đoán ung thư*”.

## 2. Mục đích nghiên cứu

Ngày nay, với số lượng hình ảnh y khoa (CT, X-Quang, MRI) ngày càng lớn thì công việc đọc và chẩn đoán hình ảnh của các bác sĩ ngày càng nặng nề, do đó việc xây dựng một mô hình phân loại để hỗ trợ các bác sĩ chẩn đoán hình ảnh là một nhu cầu cấp thiết hiện nay. Tuy nhiên để có thể xây dựng được mô hình có độ tin cậy cao đòi hỏi phải có một phương pháp tốt và lượng lớn các mẫu dữ liệu huấn luyện tức là các dữ liệu đã được gán nhãn loại tương ứng. Cho đến nay có nhiều hướng tiếp cận để giải quyết bài toán phân loại, nhận dạng đối tượng đã được đề xuất và Convolutional Neural Network (CNN - Mạng nơ-ron tích chập) là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao.

Như đã phân tích ở trên, chúng ta thấy rằng bài toán tự động phát hiện các đối tượng trong hình ảnh số là nhu cầu cấp thiết trong nhiều lĩnh vực khác nhau và bài toán phát hiện những bất thường trong hình ảnh y khoa là một bài toán có ý nghĩa thực tiễn lớn. Với mục đích nghiên cứu để giải quyết các vấn đề đã phân tích ở trên, trong luận văn này học viên đặt ra mục tiêu nghiên cứu cụ thể như sau.

### **3. Mục tiêu nghiên cứu**

- Nghiên cứu các phương pháp: Phân loại hình ảnh, phát hiện các bất thường trong hình ảnh y khoa đặc biệt tập trung vào hình ảnh X-Quang phổi.
- Thu thập và tham vấn chuyên gia để gán nhãn hình ảnh X-Quang phổi.
- Nghiên cứu mạng Neural nhân tạo và mô hình học sâu (Deep learning Model)
- Convolution Neural Network (CNN).
  - Nghiên cứu các phương pháp đánh giá mô hình phân lớp.
  - Áp dụng mô hình học sâu CNN cho bài toán phát hiện bất thường đậm độ trên các bộ dữ liệu hình ảnh X-Quang:
    - + Trên bộ dữ liệu hình ảnh do học viên tự thu thập từ bệnh viện ở TP. Hồ Chí Minh.
    - + Trên bộ dữ liệu chuẩn đã được công bố.

### **4. Đối tượng nghiên cứu và phạm vi nghiên cứu**

Nghiên cứu CNN để phát triển mô hình học sâu cho bài toán phát hiện bất thường đậm độ trên hình ảnh X-Quang phổi.

### **5. Phương pháp nghiên cứu**

Để thực hiện đề tài này, thì cần phải kết hợp hai phương pháp sau:

- Phương pháp tìm hiểu lý thuyết:
- + Mạng Neural Network.
- + Deep learning model - mô hình mạng học sâu CNN.
- Phương pháp thực nghiệm trên cơ sở dữ liệu mẫu:

Trong luận văn này, học viên sẽ nghiên cứu và sử dụng CNN để xây dựng, cài đặt mô hình để huấn luyện và phân lớp tập ảnh đầu vào và thực nghiệm đánh giá kết quả dự trên các bộ dữ liệu độc lập nhằm đánh giá hiệu năng của mô hình đã đề xuất.

### **6. Đóng góp của đề tài**

- Cấu trúc mạng CNN áp dụng cho bài toán phát hiện bất thường đậm độ trên hình ảnh X-Quang phổi.
- Trong luận văn này cũng đề xuất một mô hình mạng học sâu để phát hiện bất thường đậm độ trên hình ảnh X-Quang phổi. Mô hình này sử dụng nhiều mạng CNN

để đánh giá từng thành phần của bức ảnh và đưa ra kết luận cuối cùng là bức ảnh thuộc về lớp nào, luận văn gọi mô hình này là **Multi-CNNs**.

- Trong luận văn này cũng đề xuất một quy tắc kết hợp (Fusion rules) để tổng hợp các kết quả của các thành phần của mô hình Multi-CNNs từ đó đưa ra kết luận cuối cùng cho bức ảnh đầu vào.

- Thu thập và xây dựng bộ dữ liệu hình ảnh X-Quang phổi (*bộ ảnh đã được chuyên gia gán nhãn*).

## 7. Nội dung luận văn

Luận văn này bao gồm 5 chương:

**Chương 1:** Giới thiệu tổng quan.

Trong chương này, luận văn tập trung giới thiệu tổng quan về bài toán phân loại hình ảnh nói chung và bài toán phát hiện bất thường trên hình ảnh ý khoa; Trình bày tình hình nghiên cứu trong và ngoài nước liên quan đến nội dung mà luận văn nghiên cứu.

**Chương 2:** Cơ sở lý thuyết.

Trong chương này, luận văn tập trung trình bày cơ sở lý thuyết khi thực hiện đề tài, khái niệm về Mạng neural, CNN, phương pháp đánh giá mô hình và luận văn sử dụng.

**Chương 3:** Mô hình Multi-CNNs phát hiện bất thường đậm độ trên hình ảnh X-Quang ngực.

Trong chương này luận văn tập trung trình bày về mô hình Multi-CNNs mà học viên đã nghiên cứu và đề xuất.

**Chương 4:** Thực nghiệm và đánh giá kết quả.

Trong chương này luận văn tập trung trình bày kết quả thực nghiệm mô hình đã đề xuất ở chương 3 trên bộ dữ liệu mà học viên thu thập được từ bệnh viện và 01 bộ dữ liệu chuẩn đã được công bố.

**Chương 5:** Kết luận và hướng phát triển.

# CHƯƠNG 1 - TỔNG QUAN VỀ BÀI TOÁN PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRONG ẢNH Y KHOA

## 1.1. Tổng quan về bài toán phân lớp

### 1.1.1. Khái niệm về phân lớp

Ngày nay, cùng với sự bùng nổ như vũ bão của công nghệ thông tin và thế giới kỹ thuật số, khối lượng dữ liệu và thông tin phát triển không ngừng. Với một lượng dữ liệu và thông tin ngày càng lớn như vậy, phân lớp dữ liệu dần trở thành một nhu cầu tất yếu nhằm phục vụ cho cuộc sống. Vấn đề đặt ra là làm thế nào để nâng cao hiệu quả phân lớp và độ chính xác của quá trình phân lớp.

Phân lớp (classification) là một tiến trình xử lý nhằm xếp các mẫu dữ liệu hay các đối tượng vào một trong các lớp đã được định nghĩa trước. Các mẫu dữ liệu hay các đối tượng được xếp vào giá trị của các thuộc tính (attributes) của mẫu dữ liệu hay đối tượng đó. Sau khi đã xếp tất cả các đối tượng đã biết trước vào các lớp tương ứng, lúc này mỗi lớp được đặc trưng bởi tập các thuộc tính của các đối tượng chứa trong lớp đó.

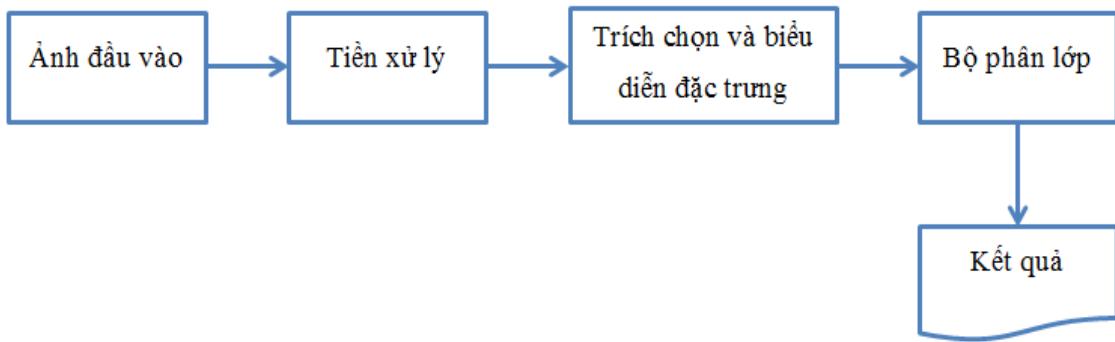
Có rất nhiều cách tiếp cận phân lớp, nhưng nói chung có 2 cách cơ bản nhất:

- Phân lớp học có giám sát (supervised learning).
- Phân lớp học không có giám sát (unsupervised learning).

### 1.1.2. Bài toán phân loại ảnh

Trong những năm gần đây, phân loại hình ảnh đã thu hút được nhiều sự quan tâm của các chuyên gia trong và ngoài nước. Và cũng đã đạt được những thành quả nhất định. Các nhà nghiên cứu đã nỗ lực rất lớn trong việc tìm ra nhiều hơn nữa các thuật toán phân loại và kỹ thuật mới, hoặc phát triển các phương pháp tiếp cận và kỹ thuật hiện có để nâng cao độ chính xác của việc phân loại.

Các bước chính của quá trình phân loại hình ảnh có thể bao gồm các bước sau: xác định một hệ thống phân loại phù hợp, lựa chọn các mẫu huấn luyện, tiền xử lý hình ảnh, rút trích đặc trưng, lựa chọn các phương pháp tiếp cận phân loại phù hợp, phân loại, và đánh giá chính xác phân loại.



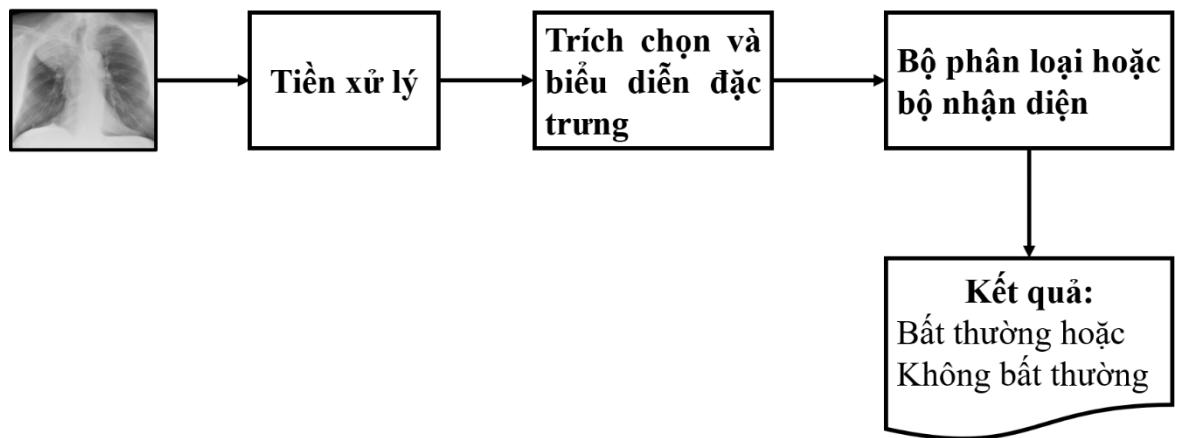
**Hình 1-1:** Mô hình bài toán phân loại hình ảnh

### 1.1.3. Bài toán phát hiện bất thường trên hình ảnh y khoa

Suy tim và phổi chiếm hơn 500.000 ca tử vong hàng năm ở Hoa Kỳ và hình ảnh X-Quang phổi thường được sử dụng để sàng lọc [13]. Ở Việt Nam và hầu hết các nước trên thế giới, hình ảnh X-Quang và CT được sử dụng nhiều trong chẩn đoán dấu hiệu ban đầu của các loại bệnh y khoa nói chung và ung thư nói riêng. Hình ảnh X-Quang ngực cung cấp một cái nhìn duy nhất của khoang ngực còn hình ảnh CT có thể cung cấp một cái nhìn toàn diện về sâu bên trong của khoang ngực và do đó hình ảnh CT có thể được sử dụng để dễ dàng phát hiện hình dạng, kích thước, vị trí và mật độ của các nốt phổi [8]. Tuy nhiên, công nghệ chụp CT khá tốn kém và thường không có sẵn trong các bệnh viện nhỏ hoặc các khu vực nông thôn. Ngược lại, công nghệ chụp hình ảnh X-Quang tương đối rẻ, nhanh và ảnh hưởng của bức xạ với bệnh nhân ít hơn CT, do đó nó thường được sử dụng trong bước chẩn đoán đầu tiên để phát hiện bất kỳ bất thường nào về ngực.

Bài toán phát hiện bất thường trong hình ảnh y khoa là bài toán có ý nghĩa thực tiễn lớn nhằm hỗ trợ cho các bác sĩ trong việc chẩn đoán bệnh trong y học, kết quả phát hiện càng chính xác thì càng hiệu quả trong tiến trình chẩn đoán và điều trị của bác sĩ. Để có thể ứng dụng các phương pháp máy học để giải quyết bài toán này, chúng ta cần có những phương pháp và mô hình có độ chính xác cao. Về bản chất trong một phạm vi hẹp theo cách đánh giá của học viên, thì bài toán phát hiện bất thường trên hình ảnh y khoa cũng là một bài toán phân loại ảnh thông thường. Do đó,

Để ứng dụng máy học cho bài toán phát hiện bất thường hay phân loại hình ảnh y khoa cũng được tiến hành theo các bước chính của bài toán phân loại ảnh.



**Hình 1-2:** Mô hình tổng quát bài toán phát hiện bất thường hình ảnh y khoa

Trong những năm gần đây, máy học đã được ứng dụng nhiều trong việc phát hiện và phân loại các hình ảnh y khoa nhằm hỗ trợ trong việc phát hiện sớm các biểu hiện của bệnh lý. Đặc biệt nếu phát triển được một hệ thống phân loại có độ chính xác cao thì nó có thể giúp bác sĩ chuyên khoa chẩn đoán hình ảnh giảm thiểu khối lượng công việc. Có rất nhiều phương pháp được đề xuất để giải quyết vấn đề này như: K-Mean, K-NN, Deep Neural Network, Support Vector Machine (SVM), ... Một hướng tiếp cận khá phổ biến trong những năm gần đây là sử dụng Convolutional neural network (CNN), đây là một mô hình học sâu thu hút được nhiều sự quan tâm của các nhà nghiên cứu trong và ngoài nước. Nó được sử dụng rất nhiều trong phân loại hình ảnh, nhận dạng hình ảnh, dịch ngôn ngữ, chẩn đoán y tế, và nhiều lĩnh vực khác, vv và đưa ra kết quả với độ chính xác cao.

## 1.2. Tình hình nghiên cứu trong nước

Tại Việt Nam trong những năm gần đây có nhiều nhóm nghiên cứu về mô hình học sâu (deep learning model) và ứng dụng trong nhiều lĩnh vực. Điển hình một số đề tài sau:

- Luận văn Thạc sỹ của ông Vũ Mạnh Hùng [3]. Trong luận văn này, tác giả thực hiện giải quyết một lớp các bài toán nhận dạng chữ viết tay mà cụ thể là nhận dạng các ký tự đơn lẻ các chữ cái từ A đến Z với 2 mô hình mô hình học sâu (deep-

learning): MLP và RBM. Tác giả kết luận: Việc áp dụng mô hình RBM cho ra một kết quả tương đối tốt, với mô hình này kết quả thu được trên bộ dữ liệu chữ viết tay tuyển sinh là 99.69%, tốt hơn xấp xỉ khoảng 2% so với mô hình MLP (97,94%).

- Luận văn Thạc sỹ của bà Lê Thị Thu Hằng [1]. Trong luận văn này, tác giả xây dựng mô hình mạng neural tích chập để nhận dạng các ký tự số trên bảng số xe. Tác giả tiến hành thử nghiệm lần lượt 376 ảnh dữ liệu về xe để kiểm tra kết quả phát hiện biển số, tách ký tự và nhận dạng ký tự, ghi nhận kết quả. Kết quả, tỉ lệ biển nhận dạng đúng vùng biển số: 372/376 ~ 98%, tỉ lệ biển tách đúng và đầy đủ ký tự: 330/335 ~ 98%. Kết quả nhận dạng từ khâu phát hiện biển số, đến tách ký tự và nhận dạng ký tự đạt xấp xỉ 65% với dữ liệu có nhiều ảnh không đạt tiêu chuẩn như bị bóng mờ, quá dơ, nhòe, ... Với ảnh biển số sạch sẽ, rõ ràng, không chứa các phụ kiện gắn trên biển, tỉ lệ nhận dạng có thể đạt tới hơn 70%.

Đối với bài toán phát hiện bát thường trong hình ảnh y khoa có luận văn Thạc sỹ của bà Nguyễn Thị Thu [2], trong đề tài tập trung nghiên cứu các kỹ thuật phát hiện đối tượng bát thường trong ảnh y học và xây dựng chương trình thử nghiệm với dựa trên kỹ thuật phân loại với phương pháp K-NN.

### 1.3. Tình hình nghiên cứu ngoài nước

Ung thư là một căn bệnh ngày càng được các giới chuyên gia quan tâm ở hầu hết các nước trên thế giới. Sử dụng phương pháp học sâu cho các bài toán phát hiện và hỗ trợ chẩn đoán ung thư là vấn đề được nhiều nhóm nghiên cứu quan tâm và phát triển. có thể liệt kê một số công trình nghiên cứu ở các nước trên thế giới về vấn đề này như sau:

- Nhóm tác giả J. D. Gallego-Posada, D. A. Montoya-Zapata, O. L. Quintero-Montoya thuộc Research Group on Mathematical Modeling School of Mathematical Sciences Universidad EAFIT Medellín, Colombia [7] trình bày một ứng dụng của Deep CNN cho việc phát hiện và chẩn đoán các khối u vú. Trong công trình này, nhóm tác giả sử dụng hình ảnh được trích xuất từ cơ sở dữ liệu mini-MIAS của nhũ ảnh. Hệ thống đề xuất được cài đặt với 3 bước. (a) cắt, xoay và thay đổi kích thước của nhũ ảnh gốc; (b) trích xuất đặc trưng sử dụng mô hình pretrained CNN. (c) sử

dụng bộ phân lớp Support Vector Machine với các đặc trưng đã được trích xuất trước đó. Trong công trình này CNN đóng vai trò như một bộ trích xuất đặc trưng và độ chính xác đạt 64.52%.

- Nhóm tác giả Daniel Lévy, Arzav Jain thuộc đại học Stanford [2] trình bày một phương pháp sử dụng CNN để phân loại trực tiếp các khối u vú lành tính hay ác tính trong nhũ ảnh sử dụng một sự kết hợp của việc học chuyển tiếp, tiền xử lý và gia tăng dữ liệu để vượt qua hạn chế về dữ liệu huấn luyện. Phương pháp này đạt được kết quả tối ưu trên bộ dữ liệu DDSM (Digital Database for Screening Mammography) với độ chính xác từ 64% đến 90%.

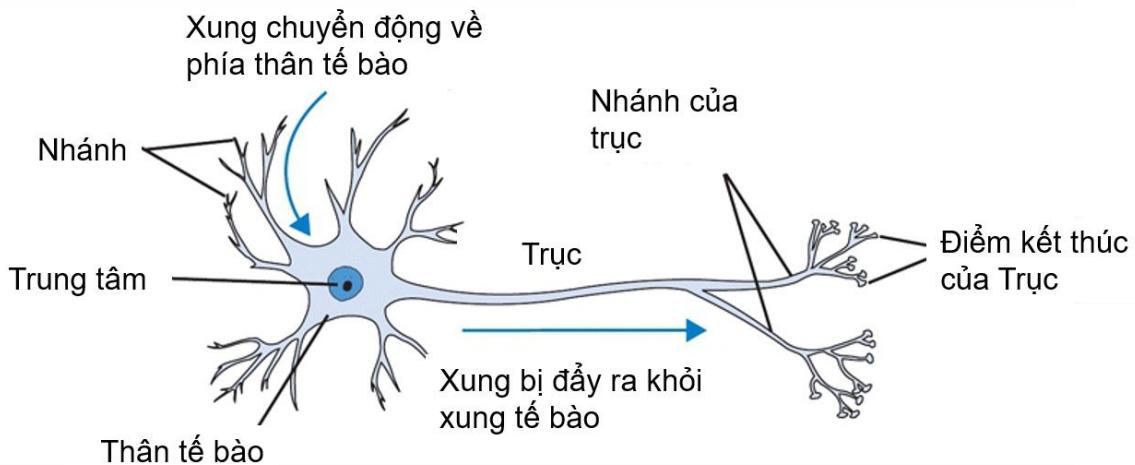
- Nhóm tác giả Omar S. Soliman, Eman AboElHamid [12] đề xuất một thuật toán phân loại lai bằng cách sử dụng Differential Evolution (DE) và Vector Vector Hỗ trợ Vùng (LS-SVM). Kỹ thuật LS-SVM được sử dụng để phân loại. Vì bộ phân loại LS-SVM rất nhạy cảm với những thay đổi của các giá trị tham số của nó, thuật toán DE được sử dụng như một kỹ thuật tối ưu cho các tham số LS-SVM. Điều này sẽ đảm bảo hiệu quả của thuật toán lai bằng cách tìm kiếm các giá trị tối ưu của phân loại. Thuật toán được đề xuất được thực hiện và đánh giá trên Bộ dữ liệu Ung thư vú ở Wisconsin (WBCD) từ Kho Cơ sở dữ liệu Học của UCI. Thuật toán được đề xuất đã được so sánh với các thuật toán phân loại khác nhau được áp dụng trên cùng một cơ sở dữ liệu. Các kết quả thực nghiệm cho thấy sự vượt trội của thuật toán đề xuất có thể đạt được độ chính xác phân loại là 99,75%.

Và một số công trình khác được đã được công bố như [6], [11], [9].

## CHƯƠNG 2 - CƠ SỞ LÝ THUYẾT

### 2.1. Mạng mạng nơron nhân tạo

Mạng nơron nhân tạo (Artificial Neural Networks - ANN) là mô hình xử lý thông tin được mô phỏng lại mạng nơron sinh học (Hình 2-1), nó là một cấu trúc khối gồm các đơn vị tính toán đơn giản được liên kết chặt chẽ với nhau trong đó các liên kết giữa các nơron quyết định chức năng của mạng. ANN giống như bộ não con người, được học bởi kinh nghiệm (qua huấn luyện), có khả năng lưu giữ những kinh nghiệm hiểu biết (tri thức) và sử dụng những tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).

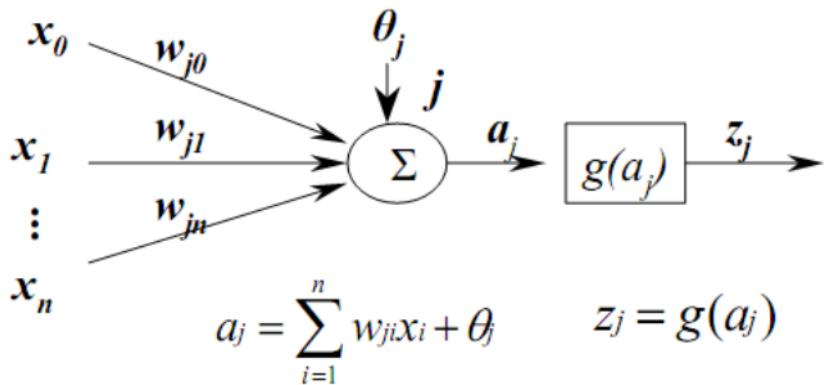


Hình 2-1: Mô hình mạng nơron sinh học

#### 2.1.1. Các thành phần cơ bản của mạng nơron nhân tạo

##### Đơn vị xử lý

Còn được gọi là một nơron hay một nút (node), thực hiện một công việc rất đơn giản: nó nhận tín hiệu vào từ các đơn vị phía trước hay một nguồn bên ngoài và sử dụng chúng để tính tín hiệu ra sẽ được lan truyền sang các đơn vị khác. Một nơron là thành phần cơ bản của một ANN. Cấu trúc của một nó được mô tả trong Hình 2-2.



Hình 2-2: Cấu trúc 1 nơron nhân tạo

Trong đó:

- +  $x_i$ : là các đầu vào ( $i = 1 \dots n$ ).
- +  $w_{ji}$ : là các trọng số tương ứng với mỗi đầu vào.
- +  $\theta_j$ : độ lệch (bias).
- +  $a_j$ : là đầu vào mạng (net - input).
- +  $z_j$ : đầu ra của nơron.
- +  $g(x)$ : hàm truyền (hàm kích hoạt).

Trong một mạng nơron có ba kiểu đơn vị:

- Các đơn vị đầu vào (Input units), nhận tín hiệu từ bên ngoài.
- Các đơn vị đầu ra (Output units), gửi dữ liệu ra bên ngoài.
- Các đơn vị ẩn (Hidden units), tín hiệu vào (input) và ra (output) của nó nằm trong mạng.

Mỗi đơn vị  $j$  có thể có một hoặc nhiều đầu vào:  $x_0, x_1, x_2, \dots, x_n$ , nhưng chỉ có một đầu ra  $z_j$ . Một đầu vào tới một đơn vị có thể là dữ liệu từ bên ngoài mạng, hoặc đầu ra của một đơn vị khác, hoặc là đầu ra của chính nó.

### **Hàm kết hợp (hợp nhất)**

Mỗi một đơn vị trong một mạng kết hợp các giá trị đưa vào nó thông qua các liên kết với các đơn vị khác, sinh ra một giá trị gọi là đầu vào mạng (net-input). Hàm thực hiện nhiệm vụ này gọi là hàm kết hợp (combination function), được định nghĩa bởi một luật lan truyền cụ thể. Trong phần lớn các mạng nơron, chúng ta giả sử rằng mỗi một đơn vị cung cấp một bộ cộng như là đầu vào cho đơn vị mà nó có liên kết. Tổng

đầu vào đơn vị  $j$  đơn giản chỉ là tổng trọng số của các đầu ra riêng lẻ từ các đơn vị kết nối cộng thêm ngưỡng hay độ lệch (bias)  $\theta_j$ :

$$a_j = \sum_{i=1}^n w_{ji}x_i + \theta_j \quad (2.1)$$

Trường hợp  $w_{ji} > 0$ , nơron được coi là đang ở trong trạng thái kích thích. Tương tự, nếu như  $w_{ji} < 0$ , nơron ở trạng thái kiềm chế. Chúng ta gọi các đơn vị với luật lan truyền như trên là các **sigma units**.

Trong một vài trường hợp người ta cũng có thể sử dụng các luật lan truyền phức tạp hơn. Một trong số đó là luật **sigma-pi**, có dạng như sau:

$$a_j = \sum_{i=1}^n w_{ji} \prod_{k=1}^m x_{ik} + \theta_j \quad (2.2)$$

Rất nhiều hàm kết hợp sử dụng một "độ lệch" hay "ngưỡng" để tính net input tới đơn vị. Đối với một đơn vị đầu ra tuyến tính, thông thường,  $\theta_j$  được chọn là hằng số.

### **Hàm truyền (hàm kích hoạt)**

Phần lớn các đơn vị trong mạng nơron chuyển đầu vào mạng (net-input) bằng cách sử dụng một hàm vô hướng (scalar-to-scalar function) gọi là hàm truyền, có rất nhiều hàm truyền có thể dùng trong các mạng neuron. Hàm truyền được phân loại theo hàm tuyến tính hoặc phi tuyến, hàm liên tục hay gián đoạn. Việc lựa chọn hàm truyền phù hợp với bài toán thường được tiến hành qua thực nghiệm. Các hàm kích hoạt hay được sử dụng được trình bày trong Bảng 2-1 dưới đây.

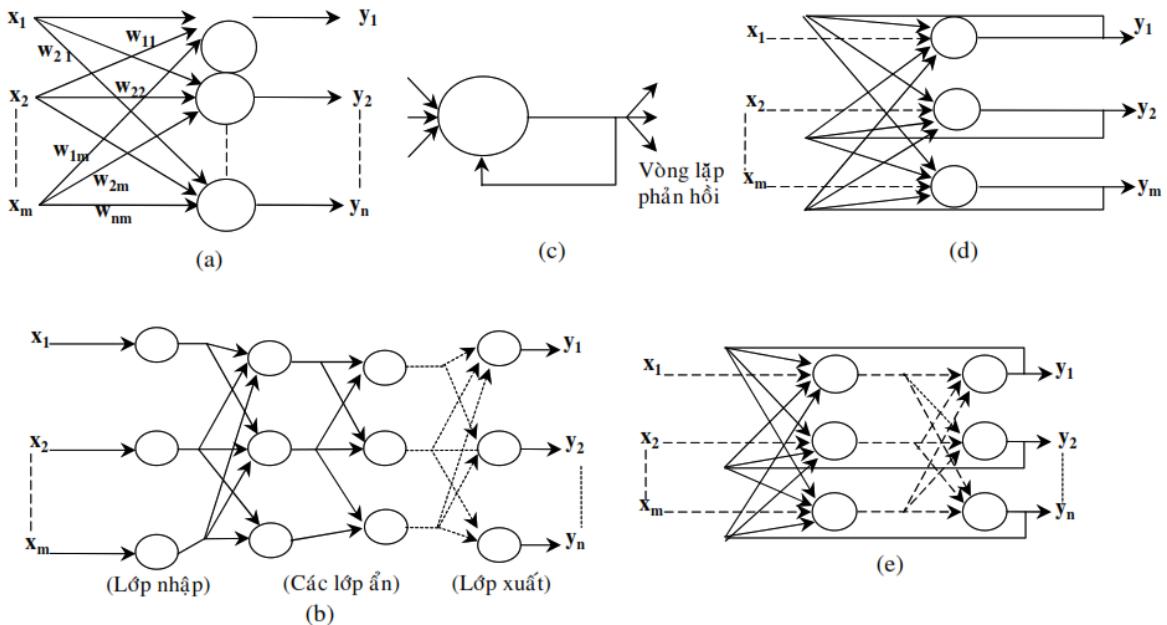
**Bảng 2-1:** Các hàm truyền cơ bản

Tên hàm	Đồ thị	Miền giá trị
Hàm đồng nhất (Linear function)		$g(x) = x$

Hard limit function		$g(x) = \begin{cases} 1, & \text{nếu } x \geq \theta \\ 0, & \text{nếu } x < \theta \end{cases}$
Hàm sigmoid		$g(x) = \frac{1}{1+e^{-x}}$
Bipolar sigmoid function (tansig)		$g(x) = \frac{1-e^{-x}}{1+e^{-x}}$

### 2.1.2. Cấu trúc mạng nơron nhân tạo – ANN

Mạng nơron nhân tạo bao gồm tập hợp các liên kết qua lại bên trong giữa các nơron trên nguyên tắc: “đầu ra của mỗi nơron được liên kết thông qua các trọng số đến các nơron khác hoặc tới chính nó. Như vậy, việc bố trí các nơron và sơ đồ liên kết qua lại giữa chúng sẽ hình thành một kiểu mạng nơron nhân tạo.



**Hình 2-3:** Một số sơ đồ liên kết của mạng nơron

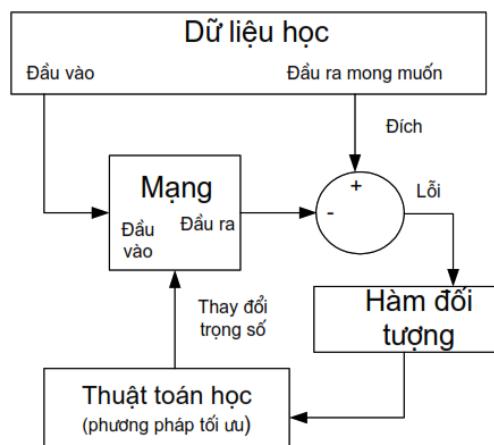
Hình 2-3 trình bày năm sơ đồ liên kết cơ bản của mạng nơron: (a) Mô hình mạng truyền thẳng một lớp; (b) mô hình mạng truyền thẳng đa lớp; (c) mô hình một nơron đơn với liên kết phải hồi đến chính nó; (d) mô hình mạng lặp một lớp; (e) mô hình mạng lặp đa lớp.

### 2.1.3. Huấn luyện mạng ANN

Chức năng của một mạng nơron được quyết định bởi các nhân tố như: cấu trúc mạng (số lớp, số đơn vị trên mỗi tầng, và cách mà các lớp được liên kết với nhau) và các trọng số của các liên kết bên trong mạng. Cấu trúc của mạng thường là cố định, và các trọng số được quyết định bởi một thuật toán huấn luyện (training algorithm). Tiến trình điều chỉnh các trọng số để mạng “nhận biết” được quan hệ giữa đầu vào và đầu ra mong muốn được gọi là học (learning) hay huấn luyện (training). Rất nhiều thuật toán huấn luyện đã được phát minh để tìm ra tập trọng số tối ưu làm giải pháp cho các bài toán. Có 3 phương pháp học phổ biến là học có giám sát (*Supervised learning*), học không có giám sát (*Unsupervised Learning*) và học tăng cường (*Reinforcement Learning*).

#### **Học có giám sát (*Supervised learning*)**

Học có giám sát (*supervised learning*) là phương pháp học từ một tập dữ liệu huấn luyện ta đã được gán trước nhãn của dữ liệu cần học và số lớp cần phân loại của bộ dữ liệu được đem đi huấn luyện. Đây là phương pháp học phổ biến nhất trong các phương pháp học của mạng neural. Mục đích của phương pháp này là từ tập dữ liệu đã gán nhãn qua quá trình học tạo ra một hàm số thật tốt để khi có dữ liệu mới ta có thể tính hay dự đoán được nhãn tương ứng của dữ liệu đó.



**Hình 2-4: Mô hình học có giám sát**

Phương pháp học giám sát chia ra thành hai loại chính là:

- Phân loại (classification): Nếu nhãn của các dữ liệu đầu vào được chia thành một số hữu hạn nhóm. Như các bài toán nhận dạng chữ số viết tay, dò tìm khuôn mặt người, ...

- Hồi quy (regression): Nếu nhãn của các dữ liệu đầu vào không được chia thành các nhóm mà là một giá trị thực cụ thể. Ví dụ như bài toán dự đoán giá của một căn nhà, giá trị dinh dưỡng của một món ăn, ...

### ***Học không có giám sát (Unsupervised Learning).***

Học không giám sát (Unsupervised Learning) (Kohonen, 2001) là phương pháp học từ một tập dữ liệu cần học ta không biết trước nhãn của dữ liệu, cũng như số lớp đầu ra. Thuật toán sẽ dựa vào cấu trúc của dữ liệu để phân nhóm hay giảm chiều của dữ liệu, ... Học không giám sát có liên quan chặt chẽ đến việc ước lượng mật độ trong thống kê. Cách học này không sử dụng tri thức từ bên ngoài trong quá trình học nên còn được gọi là tự tổ chức.

Mạng neural điển hình cho cách học này là Self - Organizing Map (SOM). Một số thuật toán học không giám sát khác như k-means, Hierarchical Agglomerative Clustering (HAC), Fuzzy Cognitive Map (FCM), .... Các hướng tiếp cận bài toán học không có giám sát là:

- Phân cụm (cluster): Chia dữ liệu thành các cụm nhỏ dựa trên sự liên quan của các dữ liệu trong mỗi cụm.

- Liên kết (association): Bài toán khám phá ra quy luật của bộ dữ liệu cho trước.

Phần lớn ứng dụng học không giám sát vào các bài toán ước lượng như mô hình hóa thống kê, nén, lọc, phân nhóm, ...

### ***Học tăng cường***

Học tăng cường (Reinforcement Learning) là phương pháp học giúp cho một hệ thống tự động xác định hành động dựa trên hoàn cảnh để đạt lợi ích cao nhất. Học tăng cường bằng cách thử nghiệm dữ liệu để khám phá ra hành động nào là tốt nhất. Hành động đó không chỉ có lợi ích tốt cho bước hiện tại mà còn cho các bước tiếp

theo. Hai đặc điểm quan trọng để phân biệt học tăng cường là tìm kiếm thử - sai (trial-and-error search) và bồi dưỡng chậm (delayed reward).

Môi trường học thường được biểu diễn dưới dạng Quy trình quyết định markov (Markov Decision Process - MDP) trạng thái hữu hạn. Và các thuật toán học tăng cường liên quan đến các kỹ thuật quy hoạch động.

Một cách hình thức, mô hình học tăng cường bao gồm: Tập các trạng thái của môi trường, tập các hành động và tập các điểm số. Học tăng cường khai thác những gì nó đã biết và khám phá môi trường mới, những hành động mới.

Hiện tại, học tăng cường chủ yếu được áp dụng vào Lý thuyết trò chơi, thuật toán cần xác định hành động tiếp theo để đạt số điểm tốt nhất. Ví dụ, AlphaGo của Google đánh cờ vây thắng con người, trong khi cờ vây có độ phức tạp cao xấp xỉ  $10^{761}$ . AlphaGo kết hợp cả học có giám sát và học tăng cường.

#### **2.1.4. Hàm mục tiêu**

Để huấn luyện một mạng và xét xem nó thực hiện tốt đến đâu, ta cần xây dựng một hàm mục tiêu (hay hàm giá) để cung cấp cách thức đánh giá khả năng hệ thống một cách không nhập nhằng. Việc chọn hàm mục tiêu là rất quan trọng bởi vì hàm này thể hiện các mục tiêu thiết kế và quyết định thuật toán huấn luyện nào có thể được áp dụng. Để phát triển một hàm mục tiêu đo được chính xác cái chúng ta muốn không phải là việc dễ dàng. Một vài hàm cơ bản được sử dụng rất rộng rãi. Một trong số chúng là hàm tổng bình phương lỗi (sum of squares error function):

$$E = \frac{1}{NP} \sum_{p=1}^P \sum_{i=1}^N (t_{pi} - y_{pi})^2 \quad (2.3)$$

Trong đó: + **p**: số thứ tự mẫu trong tập huấn luyện ( $p=1..P$ )

+ **i**: số thứ tự của đơn vị đầu ra ( $i=1..N$ )

+ **t<sub>pi</sub>** và **y<sub>pi</sub>**: tương ứng là đầu ra mong muốn và đầu ra thực tế của mạng cho đơn vị đầu ra thứ **i** trên mẫu thứ **p**.

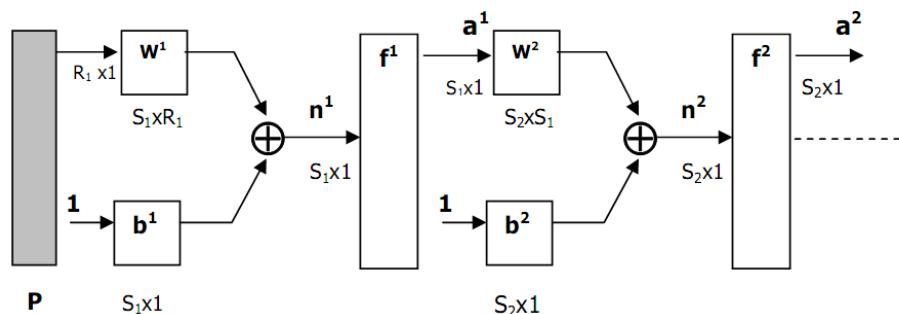
Trong các ứng dụng thực tế, nếu cần thiết có thể làm phức tạp hàm số với một vài yếu tố khác để có thể kiểm soát được sự phức tạp của mô hình.

### 2.1.5. Mạng truyền thẳng và thuật toán lan truyền ngược

Để đơn giản và tránh hiểu nhầm, mạng truyền thẳng trình bày trong phần này là mạng truyền thẳng có nhiều lớp (MLP - MultiLayer Perceptron). Đây là một trong những mạng truyền thẳng điển hình, thường được sử dụng trong các hệ thống nhận dạng.

#### Mạng truyền thẳng MLP

Một mạng truyền thẳng nhiều lớp bao gồm một lớp vào, một lớp ra và một hoặc nhiều lớp ẩn. Các nơron đầu vào thực chất không phải các nơron theo đúng nghĩa, bởi lẽ chúng không thực hiện bất kỳ một tính toán nào trên dữ liệu vào, đơn giản nó chỉ tiếp nhận các dữ liệu vào và chuyển cho các lớp kế tiếp. Các nơron ở lớp ẩn và lớp ra mới thực sự thực hiện các tính toán, kết quả được định dạng bởi hàm đầu ra (hàm chuyển). Cụm từ “truyền thẳng” (feed forward) (không phải là trái nghĩa của lan truyền ngược) liên quan đến một thực tế là tất cả các nơron chỉ có thể được kết nối với nhau theo một hướng: tới một hay nhiều các nơron khác trong lớp kế tiếp (loại trừ các nơron ở lớp ra).



Hình 2-5: Mạng nơron truyền thẳng nhiều lớp

Trong đó:

- +  $\mathbf{P}$ : Vector đầu vào (vector cột)

- +  $\mathbf{W}^i$ : Ma trận trọng số của các nơron lớp thứ  $i$ , với  $\mathbf{W}^i = \mathbf{S}_i \times \mathbf{R}_i$  ( $S$  hàng (nơron);  $R$  cột (số đầu vào))

- +  $\mathbf{b}^i$ : Vector độ lệch (bias) của lớp thứ  $i$ , với  $\mathbf{b}^i = \mathbf{S}_i \times \mathbf{1}$  (cho  $S$  nơron)

- +  $\mathbf{n}^i$ : net – input ( $\mathbf{n}^i = \mathbf{b}^i$ )

- +  $\mathbf{f}^i$ : Hàm truyền (hàm kích hoạt).

- +  $\mathbf{a}^i$ : net output.

- +  $\oplus$ : Hàm tổng thông thường.

Mỗi liên kết gắn với một trọng số, trọng số này được thêm vào trong quá trình tín hiệu đi qua liên kết đó. Các trọng số có thể dương, thể hiện trạng thái kích thích, hay âm, thể hiện trạng thái kiềm chế. Mỗi nơron tính toán mức kích hoạt của chúng bằng cách cộng tổng các đầu vào và đưa ra hàm chuyển. Một khi đầu ra của tất cả các nơron trong một lớp mạng cụ thể đã thực hiện xong tính toán thì lớp kế tiếp có thể bắt đầu thực hiện tính toán của mình bởi vì đầu ra của lớp hiện tại tạo ra đầu vào của lớp kế tiếp. Khi tất cả các nơron đã thực hiện tính toán thì kết quả được trả lại bởi các nơron đầu ra. Tuy nhiên, có thể là chưa đúng yêu cầu, khi đó một thuật toán huấn luyện cần được áp dụng để điều chỉnh các tham số của mạng.

Xét trường hợp mạng có hai lớp như Hình 2-5, công thức tính toán cho đầu ra như sau:

$$a^2 = f^2(W^2(f^1(W^1P + b^1)) + b^2) \quad (2.4)$$

Mạng có nhiều lớp có khả năng tốt hơn là các mạng chỉ có một lớp, chẳng hạn như mạng hai lớp với lớp thứ nhất sử dụng hàm sigmoid và lớp thứ hai dùng hàm đồng nhất có thể áp dụng để xấp xỉ các hàm toán học khá tốt, trong khi các mạng chỉ có một lớp thì không có khả năng này.

### **Thuật toán lan truyền ngược**

Thuật toán lan truyền ngược là dạng tổng quát của thuật toán trung bình bình phương tối thiểu (*Least Means Square-LMS*). Thuật toán này thuộc dạng thuật toán xấp xỉ để tìm các điểm mà tại đó hiệu năng của mạng là tối ưu. Chỉ số tối ưu (performance index) thường được xác định bởi một hàm số của ma trận trọng số và các đầu vào nào đó mà trong quá trình tìm hiểu bài toán đặt ra.

Bỏ qua sự phức tạp về mặt toán học, thuận toán có thể phát biểu đơn giản như sau:

#### + **Bước 1: Lan truyền xuôi các tính toán trong mạng truyền thẳng**

- Khi đó, đầu ra của một lớp trở thành đầu vào của lớp kế tiếp. Phương trình thể hiện hoạt động này như sau (trong đó M là số lớp trong mạng):

$$a^{m+1} = f^{m+1}(W^{m+1}a^m + b^{m+1}) \text{ với } m = 0, 1, \dots, M - 1 \quad (2.5)$$

- Các nơron trong lớp thứ nhất nhận các tín hiệu từ bên ngoài (với  $p$  chính là điểm bắt đầu của phương trình hình 2-5)

$$\mathbf{a}^0 = \mathbf{p},$$

- Đầu ra của lớp cuối cùng được xem là đầu ra của mạng:

$$\mathbf{a} = \mathbf{a}^M.$$

#### + **Bước 2: Lan truyền lỗi ngược lại qua mạng**

- Thuật toán lan truyền ngược sử dụng chỉ số hiệu năng là trung bình bình phương lỗi của đầu ra so với giá trị đích. Đầu vào của thuật toán chính là tập các cặp mô tả hoạt động đúng của mạng:

$$\{(\mathbf{p}_1, \mathbf{t}_1), (\mathbf{p}_2, \mathbf{t}_2), \dots, (\mathbf{p}_Q, \mathbf{t}_Q)\},$$

Trong đó  $\mathbf{p}_i$  là một đầu vào và  $\mathbf{t}_i$  là đầu ra mong muốn tương ứng, với  $i=1..Q$ .

- Mỗi đầu vào đưa vào mạng, đầu ra của mạng đối với nó được đem so sánh với đầu ra mong muốn. Thuật toán sẽ điều chỉnh các tham số của mạng để tối thiểu hóa trung bình bình phương lỗi.

#### + **Bước 3: Đánh giá lỗi học - lỗi trung bình bình phương**

- Quá trình học sẽ dừng khi sự sai khác này đạt đến 1 ngưỡng giới hạn phù hợp nào đó.

Giả sử  $E$  là độ lỗi. Nếu  $E \leq \varepsilon$  thì dừng.

#### + **Bước 4: Cập nhật lại các trọng số và độ lệch tương ứng.**

Lặp lại bước 2 cho tới khi thỏa mãn điều kiện kết thúc ở bước 3.

#### **Một số vấn đề lưu ý trong xây dựng mạng MLP [1]**

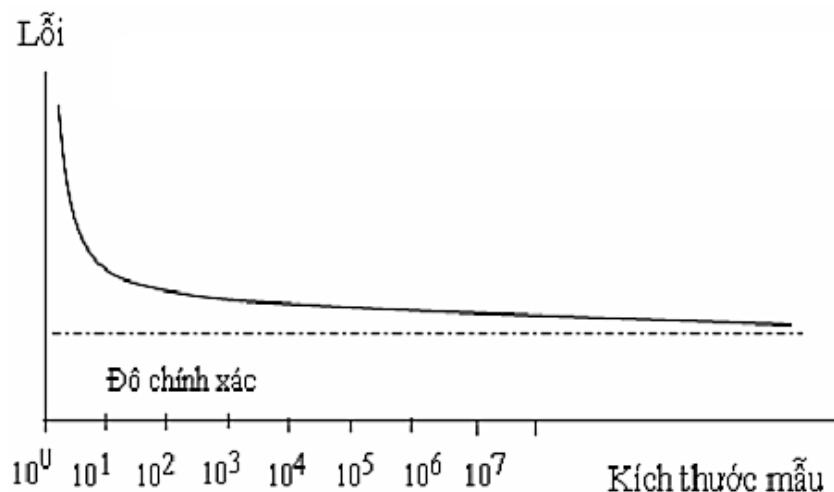
##### + **Xác định kích thước mẫu:**

Không có nguyên tắc nào hướng dẫn kích thước mẫu phải là bao nhiêu đối với một bài toán cho trước. Hai yếu tố quan trọng ảnh hưởng đến kích thước mẫu:

- Dạng hàm đích: khi hàm đích càng phức tạp thì kích thước mẫu cần tăng.

- Nghiẽu: khi dữ liệu bị nghiẽu (thông tin sai hoặc thiếu thông tin) kích thước mẫu cần tăng

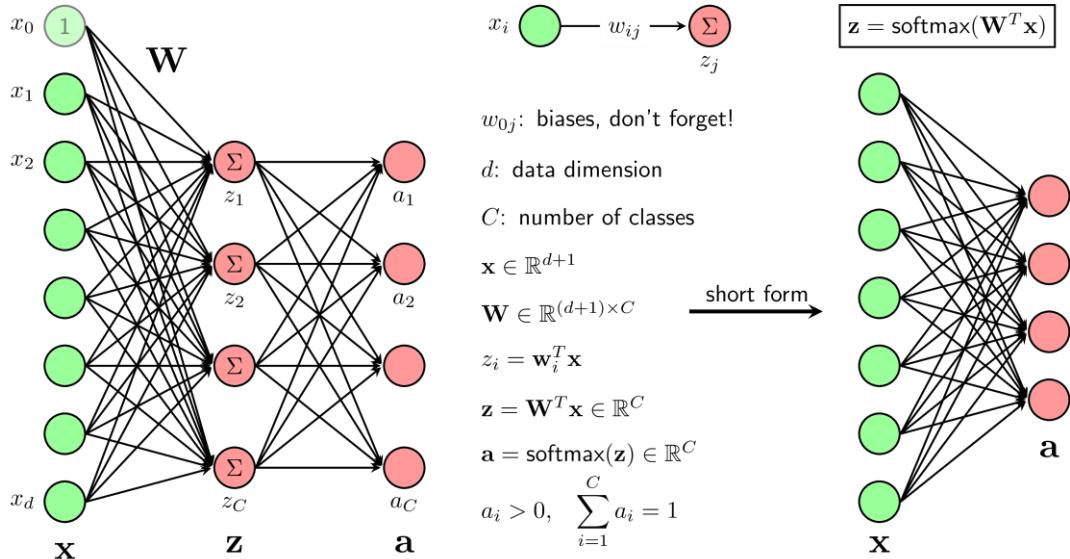
Đối với mạng truyền thẳng, cho hàm đích có độ phức tạp nhất định, kèm một lượng nhiễu nhất định thì độ chính xác của mô hình luôn có một giới hạn nhất định. Nói cách khác độ chính xác của mô hình là hàm theo kích thước tập mẫu.



**Hình 2-6:** Ảnh hưởng của kích thước mẫu

#### + Xác định số noron tầng ẩn

Câu hỏi chọn số lượng nơron trong tầng ẩn của một mạng MLP thế nào là khó, nó phụ thuộc vào bài toán cụ thể và vào kinh nghiệm của nhà thiết kế mạng. Có nhiều đề nghị cho việc chọn số lượng nơron tầng ẩn h trong một mạng MLP. Chẳng hạn **h** phải thỏa mãn  $h > (p-1)/(n+2)$ , trong đó **p** là số lượng mẫu huấn luyện và **n** là số lượng đầu vào của mạng. Càng nhiều nút ẩn trong mạng, thì càng nhiều đặc tính của dữ liệu huấn luyện sẽ được mang nắm bắt, nhưng thời gian học sẽ càng tăng.



**Hình 2-7:** Mô hình hàm softmax cho mạng Neural

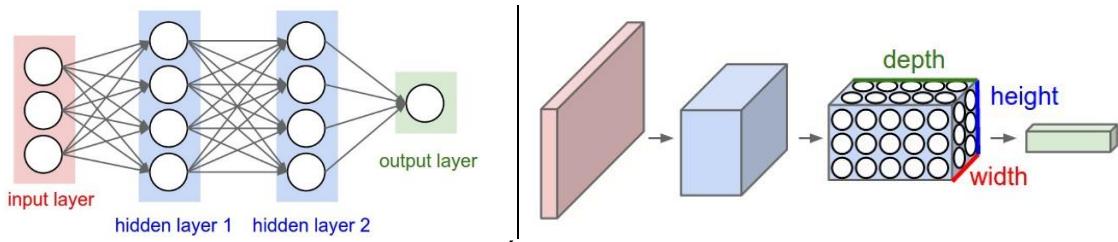
## 2.2. Kỹ thuật phân lớp ảnh sử dụng Convolutional Neural Network

Trong Những năm gần đây, ta đã chứng kiến được nhiều thành tựu vượt bậc trong ngành Thị giác máy tính (Computer Vision). Các hệ thống xử lý ảnh lớn như Facebook, Google hay Amazon đã đưa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động.

Mô hình mạng nơron truyền thẳng ra đời đã được áp dụng vào các bài toán nhận dạng. Tuy nhiên, đối với dữ liệu hình ảnh, mạng nơron truyền thẳng cho kết quả không thực sự tốt. Chính sự liên kết quá dày đủ tạo nên những hạn chế cho mô hình. Dữ liệu ảnh có kích thước khá lớn, một bức ảnh xám có kích thước  $32 \times 32$  sẽ cho ra vector đặc trưng có  $32 \times 32 = 1024$  chiều, đối với ảnh màu cùng kích thước sẽ cho ra vector đặc trưng có  $32 \times 32 \times 3 = 3072$  chiều. Với số chiều như vậy vẫn có thể quản lý được nhưng rõ ràng với cấu trúc liên kết dày đủ này khó để có thể mở rộng đến hình ảnh lớn hơn được, ví dụ với hình ảnh màu có kích thước  $128 \times 128$  sẽ cho ra  $128 \times 128 \times 3 = 49152$  chiều. Rõ ràng, kết nối dày đủ này là lãng phí và số lượng lớn các thông số sẽ nhanh chóng dẫn đến trường hợp quá khớp (over fitting).

Như đã nói ở trên, việc liên kết dày đủ các điểm ảnh trong mạng là dư thừa và lãng phí vì sự phụ thuộc lẫn nhau giữa các điểm ảnh xa nhau là không nhiều, mà chủ

yếu là sự phụ thuộc giữa các điểm ảnh lân cận nó, dựa trên tư tưởng này, mạng nơron tích chập - Convolutional Neural Networks (CNN) ra đời với cấu trúc khác hẳn mạng nơron truyền thẳng. Đầu tiên, các lớp của CNN được tổ chức theo 3 chiều: chiều rộng, chiều cao và chiều sâu và thay vì mỗi lớp được kết nối hoàn toàn với tất cả các nơron trong lớp trước đó như mạng truyền thẳng thì các nơron trong một lớp của CNN không kết nối với tất cả các nơron trong lớp tiếp theo mà chỉ đến một phần nhỏ của nó. Hình 2-7 mô tả đầy sự khác biệt cơ bản về kiến trúc của ANN và CNN.

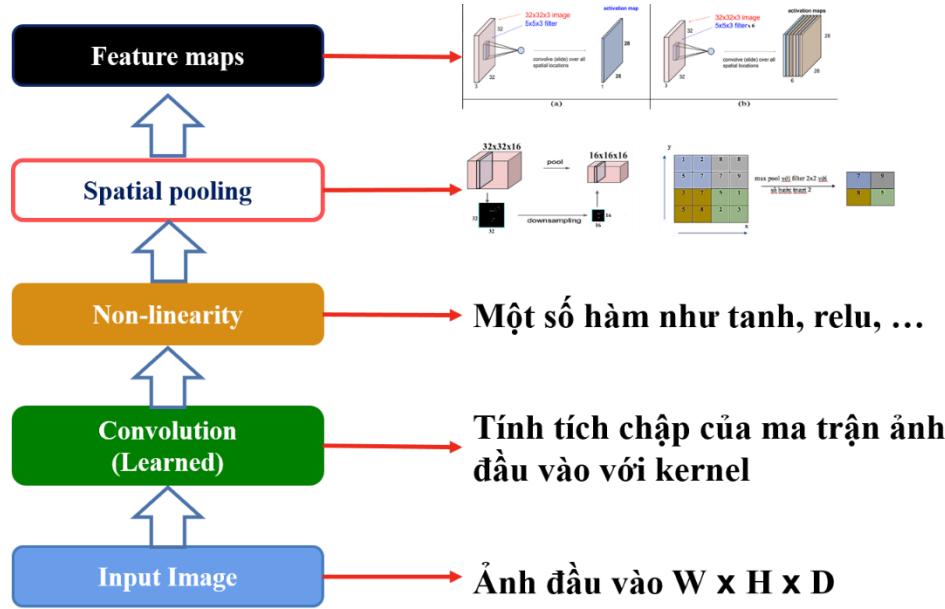


**Hình 2-7:** Kiến trúc của ANN vs CNN

CNN là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay và CNN được sử dụng nhiều trong các bài toán nhận dạng các đối tượng trong ảnh. Trong luận văn này sẽ trình bày về ý tưởng, kiến trúc của mô hình CNN cũng như áp dụng mô hình này để giải quyết bài toán phát hiện bất thường đậm đặc trên hình ảnh X-Quang phổi.

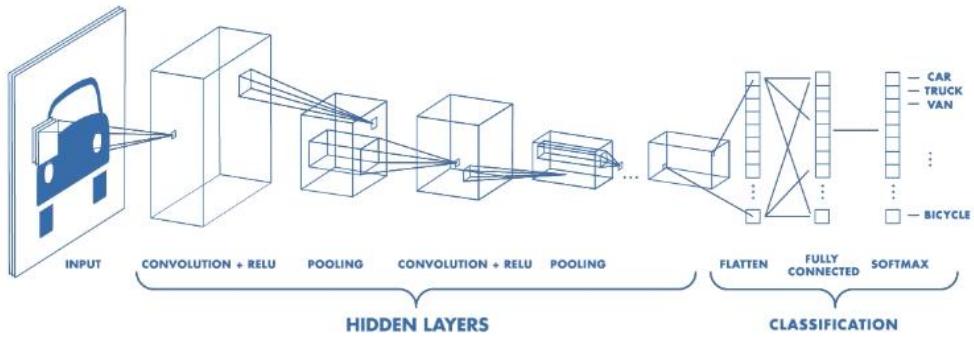
### **Kiến trúc mô hình Convolutional Neural Networks**

Các layer liên kết được với nhau thông qua cơ chế Convolution. Lớp tiếp theo là kết quả convolution từ lớp trước đó, nhờ vậy mà ta có được các kết nối cục bộ, nghĩa là mỗi nơron ở lớp tiếp theo sinh ra từ bộ lọc áp đặt lên một vùng ảnh cục bộ của các nơron lớp trước đó.



**Hình 2-8:** Cấu trúc cơ bản của mô hình CNN

Hình 2-8 mô tả cấu trúc cơ bản của mô hình CNN, từ cấu trúc trên cho thấy từ hình ảnh đầu vào qua nhiều tầng xử lý khác nhau sẽ cho kết quả xử lý cuối cùng là các bản đồ đặc trưng. Qua đây, thấy rõ ràng rằng việc trích xuất đặc trưng với mô hình CNN là hoàn toàn tự động, việc trích xuất sẽ có nhiều kết quả khác nhau dựa vào việc điều chỉnh các tầng làm việc ở giữa.



**Hình 2-9:** Kiến trúc của CNN. Nguồn: [21]

Hình 2-9 mô tả kiến trúc đầy đủ của CNN, về cơ bản kiến trúc của CNN bao gồm các lớp sau: lớp tích chập (Convolutional layer), lớp hàm kích hoạt (Activation layer), lớp pooling (Pooling layer) hay còn gọi là lớp subsampling, lớp kết nối đầy đủ (Fully Connected layer).

Trong mô hình CNN (Hình 2-9), các lớp được liên kết với nhau thông qua cơ chế tích chập (Convolution), lớp tiếp theo là kết quả của tích chập của lớp trước đó, nhờ vậy mà ta có được các kết nối cục bộ, nghĩa là mỗi nơron ở lớp tiếp theo được sinh ra từ các bộ lọc áp đặt lên một vùng ảnh cục bộ của nơron ở lớp trước đó. Mỗi lớp như vậy được áp đặt các bộ lọc khác nhau. Một số lớp khác như **pooling/subsampling** dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu). Tuy nhiên, ta sẽ không đi sâu vào khái niệm của các lớp này. Trong suốt quá trình huấn luyện, CNNs sẽ tự động học được các thông số cho các filter. Ví dụ trong tác vụ phân lớp ảnh, CNNs sẽ cố gắng tìm ra thông số tối ưu cho các filter tương ứng theo thứ tự nhất định và lớp cuối cùng được dùng để phân lớp ảnh.

CNN có tính bất biến và tính kết hợp cục bộ (*Location Invariance and Compositionalty*). Với cùng một đối tượng, nếu đối tượng này được chiếu theo các góc độ khác nhau (*translation, rotation, scaling*) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể. Lớp Pooling sẽ cho chúng ta tính bất biến đối với phép dịch chuyển (*translation*), phép quay (*rotation*) và phép co giãn (*scaling*).

Thông qua Hình 2-9, có thể hiểu cơ bản về kiến trúc của CNN đơn giản, tiếp theo sẽ làm rõ các lớp trong một kiến trúc CNN.

### **Lớp tích chập (Convolutional)**

Tích chập là một phép toán thực hiện với hai hàm số  $f$  và  $g$  với kí hiệu là  $*$ , kết quả cho ra một hàm số thứ 3. Nó được ứng dụng trong xác suất, thống kê, thị giác máy tính (computer vision), xử lý ảnh, xử lý tín hiệu, kỹ thuật điện, học máy, và các phương trình vi phân.

Tích chập của hàm số  $f$  và  $g$  được viết như sau  $f * g$ , là một phép biến đổi tích phân đặc biệt: [15]

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau = \int_{-\infty}^{\infty} f(t - \tau)g(\tau)d\tau \quad (2.8)$$

Một cách tổng quát, các hàm  $f$ ,  $g$  là hàm số phức trong không gian  $\mathbb{R}^d$ , tích chập được định nghĩa là: [15]

$$(f * g)(x) = \int_{\mathbb{R}^d} f(y)g(x-y)dy = \int_{\mathbb{R}^d} f(x-y)g(y)dy \quad (2.9)$$

Đối với Tích chập rời rạc, với hàm số phức  $f, g$  xác định trên tập số nguyên  $\mathbb{Z}$ , thì tích chập được định nghĩa: [15]

$$\begin{aligned} (f * g)(n) &= \sum_{m=-\infty}^{\infty} f(m)g(n-m)dm \\ &= \sum_{m=-\infty}^{\infty} f(n-m)g(m)dm \end{aligned} \quad (2.10)$$

Trong lĩnh vực xử lý ảnh, tích chập được sử dụng khá rộng rãi và hết sức quan trọng. Ví dụ: đối với phép lọc ảnh, phép tích chập giữa ma trận lọc và ảnh, cho ra kết quả ảnh đã được xoá nhiễu (làm mờ).

Công thức tích chập giữa hàm ảnh  $f(x, y)$  và bộ lọc  $k(x, y)$  (kích thước mxn) như sau:

$$f(x, y) * k(x, y) = \sum_{u=-\frac{m}{2}}^{\frac{m}{2}} \sum_{v=-\frac{n}{2}}^{\frac{n}{2}} k(u, v)f(x-u, y-v) \quad (2.11)$$

Trong kiến trúc của CNN, sau input là lớp tích chập (convolutional layer), tầng này chính là nơi thể hiện tư tưởng ban đầu của CNN. Thay vì kết nối toàn bộ điểm ảnh, lớp này sẽ sử dụng một bộ các bộ lọc (filters) có kích thước nhỏ so với ảnh (*thường là  $3 \times 3$  hoặc  $5 \times 5$* ) áp vào một vùng trong ảnh và tiến hành tính tích chập giữa bộ lọc và giá trị điểm ảnh trong vùng cục bộ đó.

Thành phần không thể thiếu của phép tích chập trong tầng convolutional để trích xuất đặc trưng của ảnh đầu vào là ma trận **bộ lọc** (*filter*).

**Bộ lọc (filter):** có thể thấy đây là một ma trận có kích thước  $m \times n$  như công thức 2.11 nhằm mang lại kết quả tốt nhất trên ma trận dữ liệu ảnh đầu vào.

Phép tích chập được thực hiện như sau: bộ lọc sẽ lần lượt được dịch chuyển theo một giá trị **bước trượt (stride)** bắt đầu từ góc bên trái trên của ảnh chạy dọc theo ảnh và quét toàn bộ ảnh. Ở mỗi lần dịch chuyển, thực hiện tính toán kết quả mới cho điểm ảnh đang xét bằng công thức tích chập. Có thể mô tả như sau:

Giả sử ta thực hiện phép tích chập với ảnh đầu vào là một hình ảnh 2D và ma trận bộ lọc có kích thước  $3 \times 3$ .

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

(a)
  

1	0	1
0	1	0
1	0	1

(b)

**Hình 2-10:** Biểu diễn hình ảnh đầu vào và bộ lọc dưới dạng ma trận

Hình 2-10 (a) là đầu vào cho lớp Convolution xem như hình ảnh đầu vào. Bên phải là bộ lọc (Filter) còn được gọi là hạt nhân (Kernel), tiến hành tính tích chập bằng cách trượt ma trận bộ lọc lần lượt qua tất cả các điểm ảnh trong ảnh, bắt đầu từ góc bên trái trên của ảnh. Trong mỗi lần trượt, sẽ thực hiện phép toán tích chập và đưa kết quả và ma trận kết quả. Trong hình 2-11, kết quả của phép tích chập là 4 được hiển thị trong ma trận kết quả hay bản đồ đặc trưng (Feature Map)

1x1	1x0	1x1	0	0
0x0	1x1	1x0	1	0
0x1	0x0	1x1	1	1
0	0	1	1	0
0	1	1	0	0

Input x Filter
  

4		

Feature Map

**Hình 2-11:** Tính tích chập của ảnh đầu vào với bộ lọc  $3 \times 3$

Tiếp tục thực hiện dịch chuyển ma trận bộ lọc (*trong ví dụ này bước trượt = 1*) ta được kết quả như Hình 2-12.

1	1x1	1x0	0x1	0
0	1x0	1x1	1x0	0
0	0x1	1x0	1x1	1
0	0	1	1	0
0	1	1	0	0

Input x Filter

4	3	

Feature Map

**Hình 2-12:** Thực hiện di chuyển ma trận bộ lọc

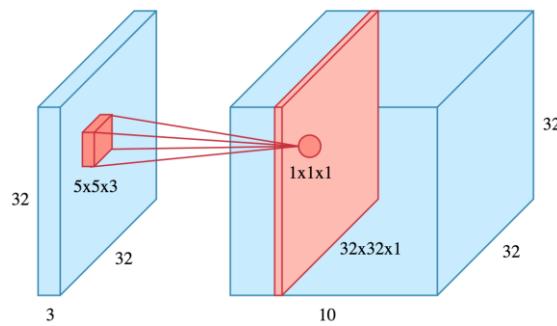
Cứ tiếp thực hiện như vậy cho đến khi tất cả các điểm ảnh của ảnh đầu vào được duyệt qua hết và ta có kết quả như Hình 2-13

1	1	1	0	0
0	1	1	1	0
0	0	1x1	1x0	1x1
0	0	1x0	1x1	0x0
0	1	1x1	0x0	0x1

4	3	4
2	4	3
2	3	4

**Hình 2-13:** Kết quả sau khi thực xong phép toán tích chập

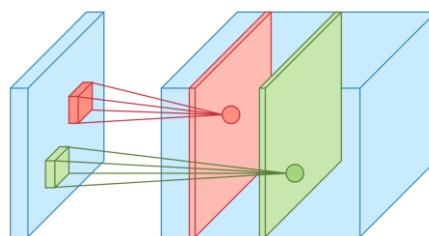
Ví dụ trên được thực hiện với hình ảnh 2D, nhưng trong thực tế nhiều bài toán đòi hỏi phải làm việc với hình ảnh 3D được biểu diễn như một ma trận 3D với kích thước: chiều rộng, chiều cao và chiều sâu (chiều sâu tương ứng với 3 kênh màu RGB). Một bộ lọc có chiều rộng và chiều cao cụ thể như  $3 \times 3$  hay  $5 \times 5$  và để áp dụng cho hình ảnh 3D cần thực hiện nhiều phép toán tích chập tương ứng với nhiều bộ lọc khác nhau để tạo ra nhiều bản đồ tính năng, sau đó gộp các bản đồ đặc trưng (feature map) này lại với nhau để trở thành đầu ra cuối cùng của lớp tích chập (convolutional layer).



**Hình 2-14:** Thực hiện phép tích chập với một bộ lọc duy nhất

Giả sử có hình ảnh đầu vào  $32 \times 32 \times 3$  và sử dụng bộ lọc có kính thước  $5 \times 5 \times 3$  (**Lưu ý:** độ sâu của bộ lọc phù hợp với độ sâu của hình ảnh). Khi bộ lọc ở một vị trí cụ thể và thực hiện phép tích chập như đã mô tả ở trên và kết quả là một bản đồ đặc trưng có kích thước  $32 \times 32 \times 1$ . Nếu sử dụng 3 bộ lọc khác nhau sẽ có 3 bản đồ đặc với kích thước  $32 \times 32 \times 1$  tương ứng và chúng được sắp xếp theo chiều sâu và cho ra kết quả cuối cùng của lớp tích chập có kích thước  $32 \times 32 \times 3$ .

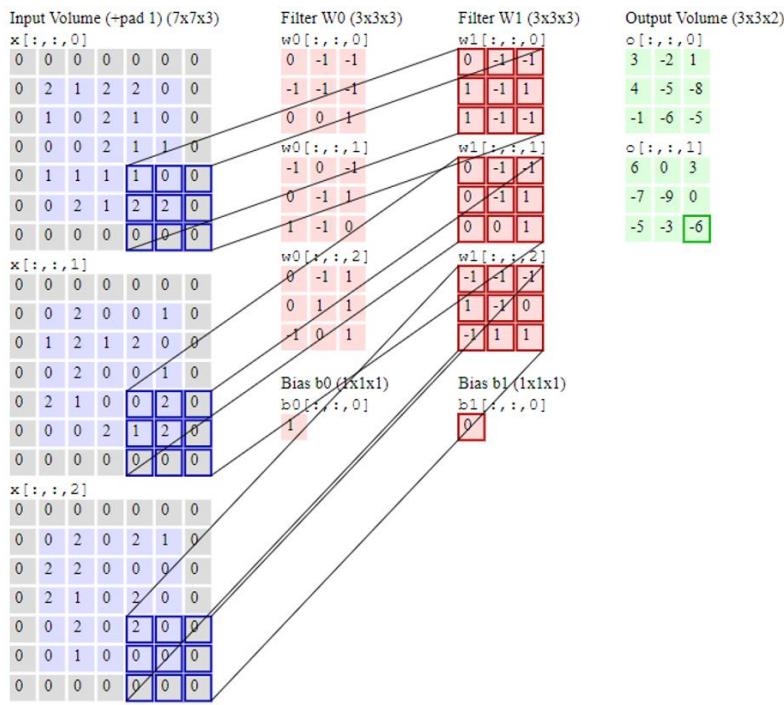
Hình 2-15 mô tả hai bản đồ đặc trưng được xếp chồng lên nhau theo chiều sâu. Hoạt động chập đối với mỗi bộ lọc được thực hiện độc lập và các bản đồ kết quả được phân tách.



**Hình 2-15:** Mô tả hai bản đồ tính năng tương ứng với hai bộ lọc

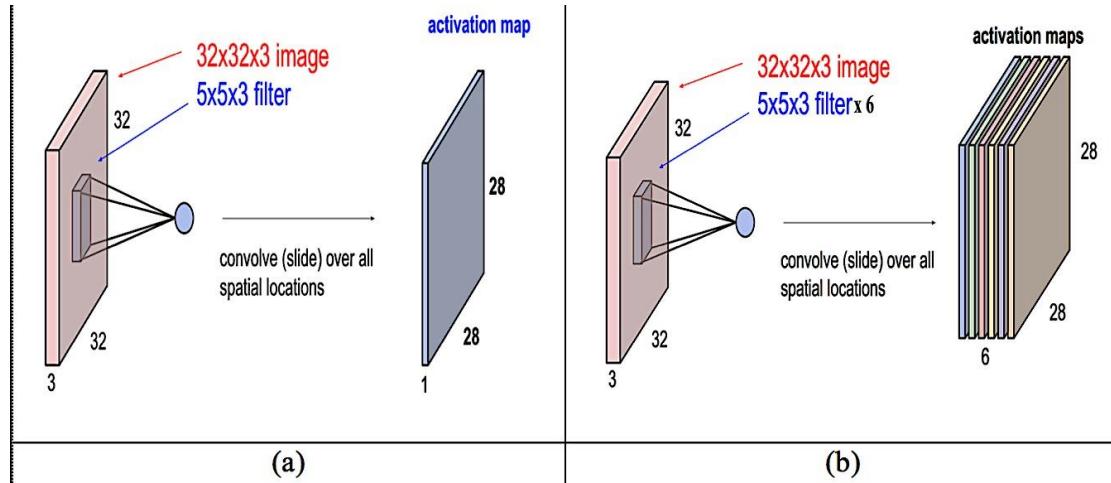
Với đầu vào là một ảnh 3 chiều  $W \times H \times 3$ , có độ sâu là 3, hay còn được gọi là 3 kênh màu, ta có tích chập 3 chiều bằng cách tính riêng tích chập cho 3 kênh này với một bộ lọc, sau đó lấy trung bình cộng của ba kết quả đầu ra, ta được tích chập 3 chiều.

Hình 2-16 minh họa cho việc tính tích chập 3 chiều. Đầu vào là một ma trận có ba kênh biểu diễn cho một ảnh 3 chiều. Ta sử dụng hai bộ lọc  $W_0$  và  $W_1$ . Cho mỗi kênh của ma trận tính tích chập với hai bộ lọc sau đó lấy trung bình cộng của ba kênh, ta được đầu ra là một ma trận có hai kênh tương ứng với hai bộ lọc.



**Hình 2-16:** Mô tả tích chập 3 chiều

Với việc thực hiện tích chập giữa ma trận dữ liệu ảnh đầu vào vào bộ lọc như đã nêu, khi đó minh họa kết quả sẽ như hình 2-17 bên dưới.

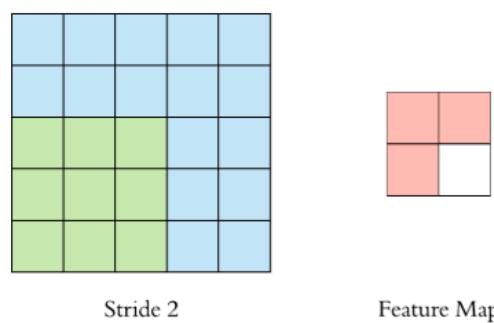


**Hình 2-17:** Mô tả hoạt động tại tầng convolution trong mô hình CNN với bộ lọc kích thước 5x5

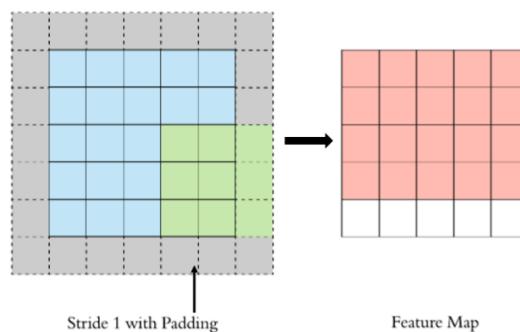
Hình 2-17 cho thấy kết quả tại tầng convolution với ảnh đầu vào là 32x32, sử dụng bộ lọc có kích thước là 5x5, kết quả sẽ là 1 bản đồ hoạt động (nhìn dưới góc độ trực quan là 1 hình ảnh đặc trưng) với kích thước 28x28 như hình 2-17.a, và bên phải

của hình 2-17 kết quả của việc sử dụng 6 bộ lọc trên cùng đầu vào, khi đó kết quả sẽ là 6 bản đồ hoạt động.

**Bước trượt (Stride) và Padding:** là kích thước của mỗi bước mà ma trận bộ lọc di chuyển trên bức ảnh ở mỗi bước tính tích chập, giá trị thông thường của nó là stride = 1 có nghĩa là sau khi tính tích chập xong tại 1 vùng ảnh, nó sẽ dịch sang phải 1 pixel, tương tự với việc dịch xuống dưới. Khi tăng kích thước của bước trượt bộ lọc sẽ di chuyển trên bức ảnh với khoảng lớn hơn do đó sẽ hạn chế được sự chồng chéo hơn giữa các điểm ảnh. Tuy nhiên việc làm này sẽ làm cho bản đồ kết quả hay bản đồ đặc trưng (feature maps) nhỏ hơn Hình 2-18. Bởi vì bản đồ tính năng luôn nhỏ hơn ảnh đầu vào, vì vậy phải tìm cách để giữ nguyên kích cỡ của feature map so với ban đầu đó là sử dụng **Padding** Hình 2-19. Khi sử dụng padding, Một lớp pixel có giá trị bằng 0 được thêm vào để bao quanh ảnh đầu vào, do đó bản đồ tính năng sẽ không co lại. Ngoài việc giữ không đổi kích thước không gian sau khi thực hiện tích chập, padding cũng cải thiện hiệu suất và đảm bảo kích thước hạt nhân và bước trượt sẽ phù hợp với đầu vào.



**Hình 2-18:** Tích chập với bước trượt = 2 (stride = 2)



**Hình 2-19:** Tích chập với bước trượt = 1 (stride = 2) và Padding = 1

Vậy kích thước đầu ra của ảnh với mỗi layer được tính như sau:

Giả sử: + Ảnh đầu vào có kích thước:  $W_1 \times H_1 \times D_1$

+ Gọi kích thước của đầu ra:  $W_2 \times H_2 \times D_2$

Thì:

$$W_2 = (W_1 - F + 2P) / S + 1$$

$$H_2 = (H_1 - F + 2P) / S + 1$$

$$D_2 = K$$

Trong đó: + F: Kích thước bộ lọc

+ S: giá trị Stride

+ P: số lượng zero-padding thêm vào ảnh đầu vào

+ K: Số lượng bộ lọc (Depth)

### Ví dụ:

Kích thước đầu ra của lớp đầu tiên trong kiến trúc CNN đã từng chiến thắng

ImageNet Chanllenge 2012 là:

Đầu vào: [227 x 227 x 3], F=11, S=4, P=0, and K=96

Đầu ra:  $(227 - 11) / 4 + 1 = 55$

→ Vậy kích thước đầu ra của lớp đầu tiên là [55 x 55 x 96]

### Lớp hàm kích hoạt

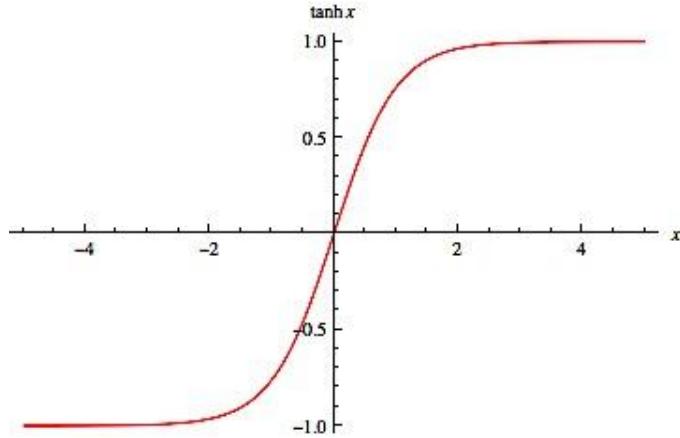
Sau mỗi lớp tích chập thường được đặt một lớp hàm kích hoạt phi tuyến để tạo ra thông tin trừu tượng hơn cho các lớp sau. Hàm kích hoạt thường được sử dụng là hàm **Tanh**, **Sigmoid**, **Rectified linear unit (ReLU)**. Nói một cách đơn giản, lớp này có nhiệm vụ chuyển toàn bộ giá trị âm trong kết quả có được từ lớp tích chập thành giá trị không âm.

+ **Hàm ReLU**: Là một hàm số tuyến tính có công thức như sau:

$$f(x) = \max(0, x) \quad (2.12)$$

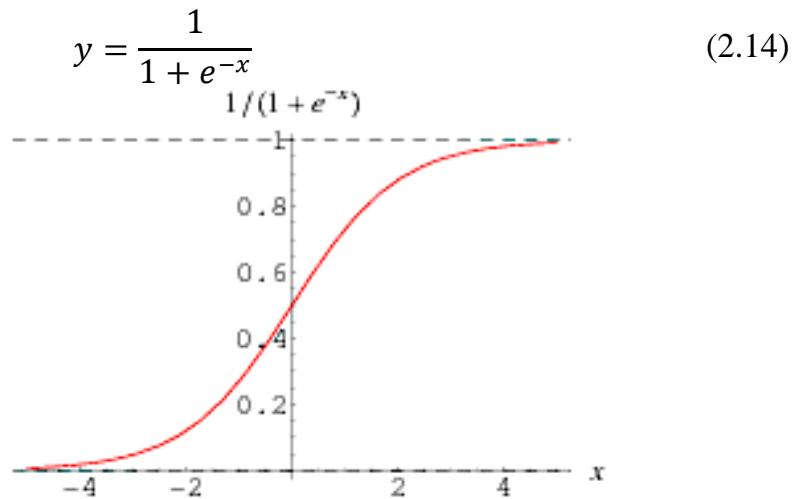
+ **Hàm Tanh**: là một hàm phi tuyến tính được định nghĩa như công thức sau:

$$\tanh \theta = \frac{\sinh \theta}{\cosh \theta} \quad (2.13)$$



**Hình 2-20:** Biểu diễn của công thức 2.13 trên hệ trục tọa độ

+ **Hàm Sigmoid:** là hàm phi tuyến tính được định nghĩa như công thức sau:

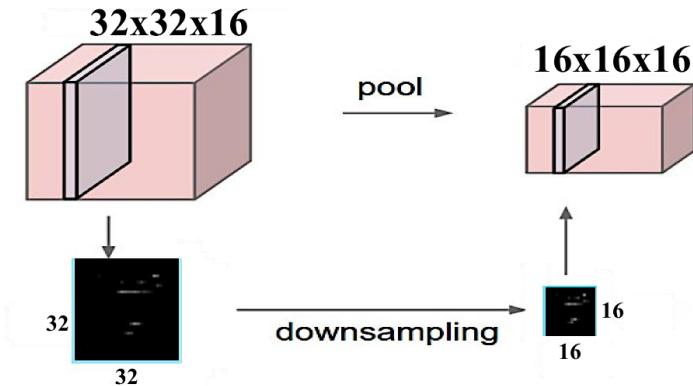


**Hình 2-21:** Biểu diễn của công thức 2.14 trên hệ trục tọa độ

### Lớp Pooling

Lớp pooling sử dụng một cửa sổ trượt quét qua toàn bộ ảnh, mỗi lần trượt theo các bước trượt cho trước. Lớp pooling không tính tích chập mà tiến hành lấy mẫu con. Khi cửa sổ trượt trên ảnh, chỉ có một giá trị được xem là giá trị đại diện cho thông tin ảnh tại vùng đó được giữ lại, giá trị này gọi là giá trị mẫu. Ta cũng có thể chia ảnh thành các ô hình chữ nhật không chồng lấp nhau để lấy mẫu.

Lớp pooling thể hiện tính bất biến đối với phép dịch chuyển, phép quay và phép co giãn của CNN. Lớp này thường được sử dụng ngay sau lớp hàm kích hoạt, để đơn giản hóa thông tin đầu ra, giảm bớt số lượng nơron, giảm bớt lượng tính toán, và do đó cũng giảm bớt hiện tượng quá khớp (*over-fitting*).



**Hình 2-22:** Minh họa về tầng pool và downsampling

Các phương thức lấy phổ biến trong lớp pooling là **max pooling** (*lấy giá trị lớn nhất*, *Hình 2-23*), **min pooling** (*lấy giá trị nhỏ nhất*), **average pooling** (*lấy giá trị trung bình*) và **L2 Pooling**, trong thực tế cả max pooling và L2 pooling được sử dụng khá rộng rãi.

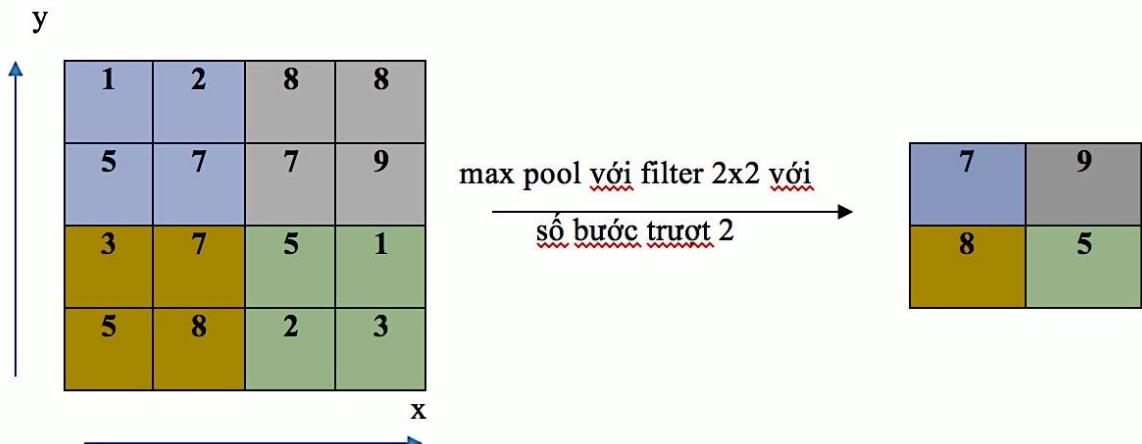
Về mặt lý thuyết với ma trận đầu vào có kích thước:  $W_1 \times H_1 \times D_1$ . Thực hiện toán tử pooling trên ma trận con của ma trận đầu vào có kích thước  $F \times F$  với bước trượt  $S$  thì ta được ma trận đầu ra  $W_2 \times H_2 \times D_2$  với các giá trị  $W_2, H_2, D_2$  được tính như sau:

$$W_2 = (W_1 - F)/S + 1$$

$$H_2 = (H_1 - F)/S + 1$$

$$D_2 = D_1$$

Max pooling lấy giá trị kích hoạt tối đa của một vùng  $m \times n$  (*trong hình 2-23 là  $2 \times 2$* ). L2 pooling thay vì lấy giá trị kích hoạt tối đa thì lấy căn bậc hai của tổng các bình phương của kích hoạt trong vùng  $2 \times 2$ . Trong chi tiết thì khác nhau, nhưng về trực giác thì tương tự như max pooling, L2 pooling là một cách để cô đọng thông tin từ các lớp tích chập.



**Hình 2-23:** Minh họa tầng pool sử dụng hàm ReLU, phương pháp Max pooling và bộ lọc kích thước  $2 \times 2$

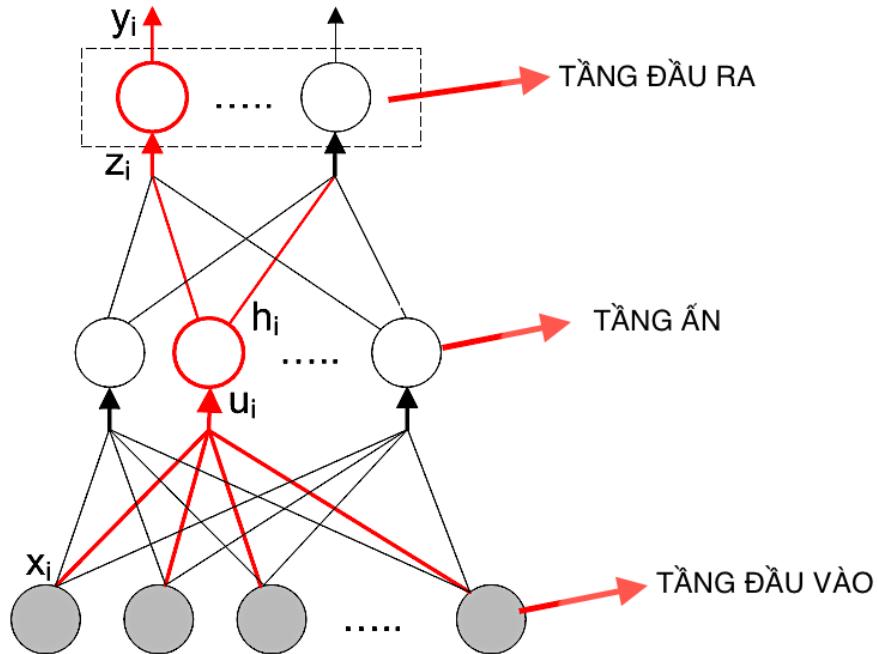
Ví dụ ở hình 2-23 có thể được mô tả như sau:

- Với mẫu đầu vào là ma trận  $4 \times 4$  (bên trái hình 2-23), sử dụng bộ lọc  $2 \times 2$  với bước trượt 2 đơn vị trên ma trận  $4 \times 4$  đầu vào áp dụng hàm ReLU.
- Lần trượt thứ nhất, sẽ áp dụng hàm ReLU trên tập các giá trị  $(1, 2, 5, 7)$  cho kết quả là 7. Lần trượt thứ 2, cho kết quả là 9. Lần trượt thứ 3 cho kết quả là 8. Lần trượt thứ 4 cho kết quả là 5. Kết quả cuối cùng được ma trận  $2 \times 2$  (bên phải hình 2-23) với các giá trị lần lượt là 7, 9, 8, 5.

### Lớp kết nối đầy đủ (*Fully Connected – FC*)

Đây là lớp cuối cùng của mô hình CNN, lớp này đóng vai trò phân lớp dữ liệu dựa trên dữ liệu đã được xử lý trước đó. Tầng FC sẽ sử dụng kết quả của các tầng trước để xuất ra một vector có độ dài là  $N$ , với  $N$  là số các lớp cần phân loại hay nhận diện. Ở lớp này, thường dùng hàm softmax để phân loại dữ liệu tương tự như trong mạng neural truyền thống.

### Softmax Classifier:



**Hình 2-24:** Mô tả các giá trị trong mô hình CNN sử dụng để xác định hàm softmax

### Công thức của hàm softmax:

Hình 2-24 trên cho thấy, nếu trong neural thứ i ở tầng output ta có:

$$y_i = \frac{e^{z_i}}{\sum_{j \in \text{group}} e^{z_j}} \quad (2.15) [4]$$

Trong đó:  $z_i$  là phần *logit* (tổng input) của mỗi nơron

Khi dùng softmax cho output layer thì hàm cost function thường được dùng là hàm cross-entropy:

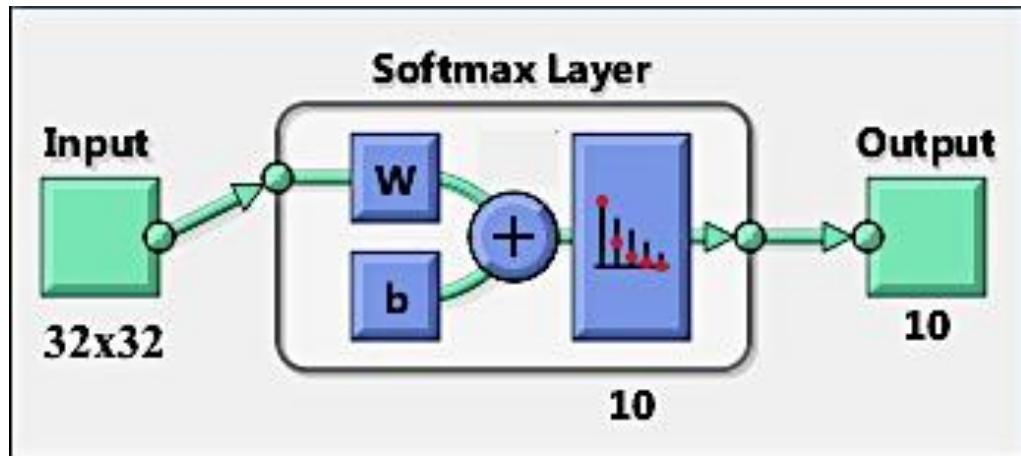
$$C = - \sum_j t_j \log y_i \quad (2.16) [4]$$

Trong công thức (2.16) [4] phía trên  $y_i$  là output của nơron thứ j theo công thức trên, và  $t_j$  là giá trị target của neural j khi huấn luyện.

Trong phần này, sử dụng bộ dữ liệu MNIST để minh họa cho tầng fc như sau, bộ dữ liệu này có 10 ký tự từ (0-9) nên tại tầng fc sẽ xuất ra một vector có chiều dài 10, nội dung ví dụ như sau: [0 .1 .1 .75 0 0 0 0 0 .05], khi đó ta có thể miêu tả như hình bên dưới.

**Bảng 2-2:** Mẫu kết quả tại tầng FC với bộ dữ liệu MNIST

Đối tượng	0	1	2	3	4	5	6	7	8	9	
[	0	0.1	0.1	0.75	0	0	0	0	0	0.05	]
Nghĩa:	0%	10%	10%	75%	0%	0%	0%	0%	0%	5%	



**Hình 2-25:** Minh họa tại tầng softmax của mô hình CNN với bộ dữ liệu MNIST

Khi đó có thể kết luận rằng tại tầng FC, với tham số đầu vào của nó là các tầng phía trước sẽ cho ra một kết quả là các xác suất rằng 10% đối tượng này số 1, 10% đối tượng này là số 2, 5% đối tượng này là số 9 và 75% đối tượng này là số 3. Đây là một minh họa cho giá trị đầu ra đối với việc sử dụng phương pháp softmax. Nhờ vào kết quả này mà ta có thể xác định được đầu ra của mô hình CNN. Có thể nói rằng tầng fc là một tầng nhìn vào các đặc trưng mức cao (*được sinh ra từ các tầng trước đó*) để xác định ra được xác xuất nhóm đối tượng.

Các đặc trưng mức cao này có thể hình dung như sau, đối với một đặc trưng mức cao ở lớp trước đó tầng FC sẽ nhận định rằng nó liên quan đến một lớp nào đó. Ví dụ nếu tầng FC dự đoán rằng đối tượng là một con chim, thì nó sẽ được đánh giá cao trong bản đồ kích hoạt thể hiện các đặc trưng cấp cao như cánh và mỏ, ...

### Lớp huấn luyện

Lớp huấn luyện lấy mạng và các tham số, sau đó truyền các mẫu đã được gán nhãn chính xác, kế đến chuyển các mẫu thông qua mạng đồng thời xem các dự đoán mạng và sau đó điều chỉnh trọng số mạng để làm cho các nhãn chính xác đã cung cấp có nhiều khả năng hơn cho đầu vào cụ thể nào đó trong tương lai. Nếu làm điều này

nhiều lần và lần lượt cho toàn bộ tập dữ liệu theo thời gian, nó sẽ thay đổi mạng để ánh xạ tất cả các đầu vào để các kết quả đầu ra chính xác.

Có 3 phương pháp huấn luyện phổ biến được dùng hiện nay:

**- SGD, SGD+Momentum:**

Để có thể hiểu SGD (Stochastic Gradient Descent) trước tiên cần hiểu được GD (Gradient Descent), cách đơn giản nhất để hiểu GD là từ một vị trí hiện tại, đi theo chiều giảm của đạo hàm bậc nhất cho đến khi không thể giảm được nữa. Khi đó, một điểm tối ưu cục bộ được tìm thấy.

Công thức cập nhật cho GD như sau:

$$\theta_{k+1} = \theta_k - \alpha_k \frac{\partial}{\partial \theta} C(\theta_k) \quad (2.17) [1]$$

Với  $\theta_k$  là tham số tại lần lặp thứ k và  $\alpha_k$  gọi là **learning rate**, có thể được cố định hoặc thay đổi thích nghi trong suốt quá trình huấn luyện.

Khi đó có thể hiểu về biến thể SGD là hướng cập nhật cho tham số là một biến ngẫu nhiên mà kì vọng của nó là hướng cập nhật tính bởi Gradient descent

Cụ thể trong tập huấn luyện thường có tính chất phân phối độc lập và phân tán nên tại mỗi bước có thể cập nhật tham số với mỗi mẫu trong tập huấn luyện:

$$\theta_{k+1} = \theta_k - \alpha_k \frac{\partial}{\partial \theta} L(\theta_k, z) \quad (2.18) [1]$$

Với  $z$  là sample tiếp theo trong tập huấn luyện, hoặc trong các ngữ cảnh *online* khi dữ liệu huấn luyện được đưa đến từng mẫu một (có thể vô hạn), và ta không có trọn vẹn cả tập huấn luyện ngay từ đầu.

SGD thông thường nhanh hơn Gradient descent vì ta cập nhật các tham số nhiều hơn hẳn. Điều này đặc biệt đúng khi ta có tập huấn luyện lớn hoặc không có toàn bộ tập huấn luyện ngay từ đầu.

**Momentum:** ý tưởng của phương pháp momentum là tính toán mức độ thay đổi của tham số tại mỗi bước dựa vào bước trước đó. Như vậy tại mỗi bước tham số sẽ thay đổi một cách “thích nghi” dần dần với các lần lặp trước. Cụ thể có thể hiểu qua công thức sau:

$$\Delta\theta_{k+1} = \varepsilon\Delta\theta_k + (1 - \varepsilon)\frac{\partial}{\partial\theta}L(\theta_k, z) \quad (2.19) [1]$$

với  $\varepsilon$  là hyperparameter điều khiển mức độ ảnh hưởng của gradient vào mức giảm tại mỗi bước.

Vậy khi kết hợp SGD và Momentum có nghĩa là tại mỗi bước sẽ cập nhật thông số trong tập huấn luyện, tuy nhiên việc cập nhật tại bước sau sẽ có tính chất ảnh hưởng bởi giá trị ở bước trước đó.

Minibatch gradient descent (MDG): nếu như SG sử dụng toàn bộ tập huấn luyện cho mỗi bước, trong khi SGD sử dụng 1 mẫu cho mỗi bước thì một hình thái mới trong cách huấn luyện được quan tâm đó là MGD, điểm khác biệt chính là có thể sử dụng  $N$  số mẫu cho mỗi bước cập nhật. Lợi thế của MGD thể hiện ở việc giảm tính ngẫu nhiên của SGD, cùng với việc giúp thuật toán hội tụ nhanh hơn. Tuy nhiên nếu  $N$  càng lớn thì thuật toán sẽ tốn chi phí cho việc tính hướng giảm tại mỗi bước.

Về giá trị **Learning Rate** có thể mô tả như sau: Learning rate là tham số kiểm soát mức độ điều chỉnh trọng số của mạng với độ loss gradient, giá trị càng thấp thì tốc độ học càng chậm, nghĩa là nếu learning rate quá nhỏ thì thuật toán sẽ hội tụ chậm. Công thức sau đây thể hiện mối quan hệ:

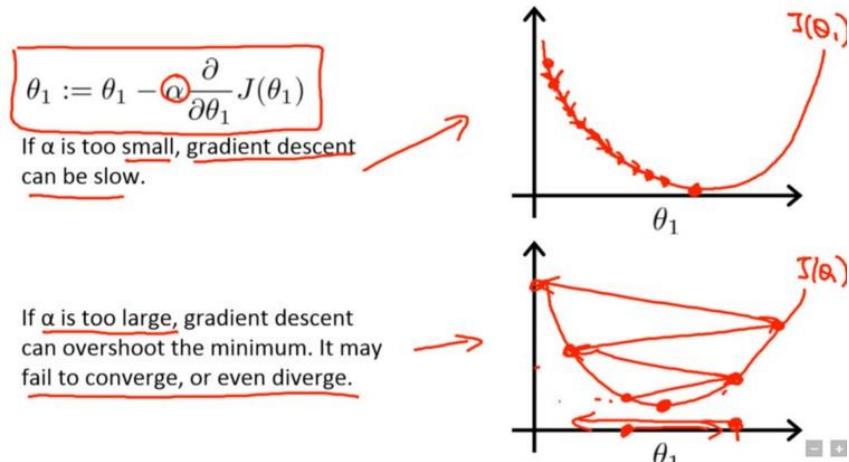
$$\mathbf{W}_{\text{new}} = \mathbf{W}_{\text{exist}} - lr * \mathbf{g} \quad (2.20) [16]$$

Trong đó:  $\mathbf{W}_{\text{new}}$ : trọng số mới sau khi được điều chỉnh.

$\mathbf{W}_{\text{exist}}$ : trọng số đã có (trong lần chạy trước đó).

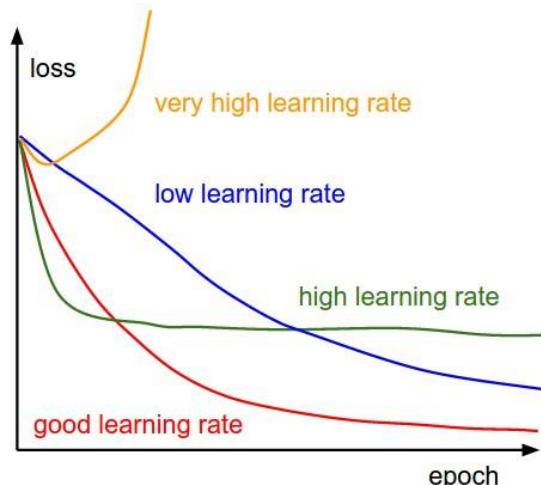
$lr$ : giá trị learning rate.

$\mathbf{g}$ : giá trị gradient descent.



**Hình 2-26:** Gradient gốc với tỷ lệ học tập nhỏ (trên) và lớn (dưới). Nguồn Andrew Ng's Machine Learning course on Coursera [16]

Thông thường learning rate được cấu hình một cách ngẫu nhiên bởi người sử dụng. Người dùng sẽ tận dụng các kinh nghiệm hoặc dựa vào các tài liệu để đạt được lựa chọn một giá trị learning rate tốt nhất. Do đó, thường khó có thể có được một giá trị learning rate đúng nhất. Biểu đồ dưới đây minh họa các kịch bản khác nhau mà người dùng có thể rơi vào khi cấu hình learning rate:



**Hình 2-27:** Ảnh hưởng của giá trị learning rate đến độ hội tụ của mô hình mạng [16]

#### - Adagrad:

Thuật toán Adagrad [16] có thể hiểu ý tưởng chính của thuật toán này như sau: Là một thuật toán dựa trên gradient descent, Nó điều chỉnh learning rate cho các tham số  $\theta$ , thực hiện các cập nhật nhỏ hơn (nghĩa là learning rate thấp) cho các thông số

liên quan đến các đặc trưng thường xuyên xảy ra và cập nhật lớn hơn (nghĩa là learning rate cao) cho các thông số liên quan đến các đặc trưng không thường xuyên, adagrad coi learning rate cũng là một tham số. Vì lý do này, nó rất phù hợp để xử lý dữ liệu thừa thớt. Dean et al. [3] đã phát hiện ra rằng Adagrad cải thiện đáng kể tính mạnh mẽ của SGD và sử dụng nó để đào tạo mạng thần kinh quy mô lớn tại Google.

Adagrad chia learning rate với tổng bình phương của lịch sử biến thiên (đạo hàm) theo công thức:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \varepsilon}} \odot g_t \quad (2.21) [17]$$

Trong đó:  $\varepsilon$  là hệ số để tránh lỗi chia cho 0, mặc định là  $1e - 8$ .

$G_t$  là một ma trận đường chéo (diagonal matrix) nơi mà mỗi phần tử  $(i, i)$  là bình phương của đạo hàm vector tham số tại thời điểm  $t$ .

Một lợi ích dễ thấy của Adagrad là tránh việc điều chỉnh learning rate bằng tay, thường sẽ để mặc định là 0.01 và thuật toán sau đó sẽ tự động điều chỉnh.

Một điểm yếu của Adagrad là tổng bình phương biến thiên sẽ lớn dần theo thời gian cho đến khi nó làm learning rate cực kì nhỏ, làm việc huấn luyện trở nên đóng băng nghĩa là tại thời điểm đó không thể học thêm được một kiến thức nào nữa.

#### - **Adadelta:**

Thuật toán Adadelta [15] dựa trên cảm hứng là sửa đổi những khuyết (việc *làm thay đổi learning rate theo tính đơn điệu giảm*) của thuật toán Adagrad nhằm cải thiện sự hội tụ của thuật toán, chú trọng 2 điểm chính so với thuật toán Adagrad:

- + Điều chỉnh learning rate dựa vào số liệu gradient trước đó chứ không phải toàn bộ quá trình trước như Adagrad.
- + Sử dụng một bộ phận để thực hiện việc tăng tốc cập nhật các thông số.

### 2.3. **Hàm Softmax**

Hàm softmax là một hàm kích hoạt, thường được sử dụng ở lớp cuối cùng để phân loại. Chúng ta cần một mô hình xác suất sao cho với mỗi điểm dữ liệu đầu vào  $\mathbf{x}, a_i$ , thể hiện xác suất để dữ liệu đầu vào đó rơi vào lớp thứ  $i$ . Điều kiện cần là các  $a_i$  phải dương và tổng của chúng phải bằng 1. Để có thể thỏa mãn điều kiện này ta

phải nhìn vào các  $z_i$  và dựa trên quan hệ giữa các  $z_i$  để tính toán các giá trị  $a_i$ . Giá trị  $z_i$  càng lớn thì xác suất dữ liệu rơi vào lớp thứ  $i$  càng cao, điều này chỉ ra rằng chúng ta cần một hàm đồng biến ở đây.

Hàm softmax được định nghĩa như sau:

$$a_i = \frac{e^{z_i}}{\sum_{j=1}^C e^{z_j}}, \quad \forall i = 1, \dots, C \quad (2.22)$$

Giá trị  $z_i$  có thể nhận giá trị âm hoặc dương. Vì vậy hàm tron đơn giản  $e^{z_i}$  có thể chắc chắn biến  $z_i$  thành một giá trị dương, và hơn nữa còn là hàm đồng biến. Để đảm bảo tổng các  $a_i$  luôn bằng một, ta đem chia  $e^{z_i}$  cho  $\sum_{j=1}^C e^{z_j}$ . Từ đó, ta có được định nghĩa về hàm softmax.

Hàm softmax, tính tất cả các  $a_i$  dựa vào tất cả các  $z_i$  thỏa mãn tất cả các điều kiện đã xét là dương, tổng bằng 1, và giữ được thứ tự của  $z_i$ . Với định nghĩa này, không có xác suất  $a_i$  nào hoàn toàn bằng 0, hoặc hoàn toàn bằng 1, mặc dù chúng có thể rất gần 0 hoặc 1 khi  $z_i$  rất nhỏ hoặc rất lớn.

Lúc này, ta có thể giả sử rằng:

$$P(y_k = i | \mathbf{x}_k; W) = a_i \quad (2.23)$$

trong đó,  $P(y = i | \mathbf{x}; W)$  được hiểu là xác suất để điểm dữ liệu  $\mathbf{x}$  rơi vào lớp thứ  $i$  nếu biết tham số mô hình  $W$ .

## 2.4. Vấn đề quá khớp (over-fitting)

**Khái niệm quá khớp:** là hiện tượng mô hình tìm được quá khớp với dữ liệu huấn luyện, nên nó sẽ trả lời chính xác những gì đã được học, còn những gì không được học thì nó không quan tâm. Như vậy mô hình sẽ không có được khả năng tổng quát hóa và chất lượng mô hình không còn tốt như khi thực hiện trên dữ liệu kiểm tra nữa.

Vấn đề quá khớp xảy ra khi mô hình quá phức tạp để mô phỏng dữ liệu huấn luyện. Điều này đặc biệt xảy ra khi lượng dữ liệu huấn luyện quá nhỏ trong khi độ phức tạp mô hình quá cao. Có 3 cách để hạn chế bớt độ phức tạp của mô hình:

- + Hạn chế số tầng ẩn

- + Ngăn không cho mô hình sử dụng các trọng số lớn.
- + Giới hạn số bước huấn luyện

Khi mô hình được huấn luyện, nó chuyển từ các hàm ánh xạ tương đối đơn giản đến các hàm ánh xạ tương đối phức tạp. Nó sẽ đạt được một cấu hình tổng quát hóa tốt nhất tại một điểm nào đó. Sau điểm đó mô hình sẽ học để mô hình hóa nhiều, những gì mô hình học được sẽ trở thành quá khớp. Nếu ta phát hiện ra thời điểm mạng đạt đến trạng thái tốt nhất này, ta có thể ngừng tiến trình huấn luyện trước khi hiện tượng quá khớp xảy ra.

Ta biết rằng, chỉ có thể để đánh giá mức độ tổng quát hóa của mô hình bằng cách kiểm tra mô hình trên các mẫu nó không được học. Ta thực hiện như sau: chia mẫu thành 02 tập mẫu huấn luyện và tập mẫu kiểm tra. Luyện mạng với tập mẫu huấn luyện nhưng định kỳ dừng lại và đánh giá sai số trên tập mẫu kiểm tra. Khi sai số trên tập mẫu kiểm tra tăng lên thì quá khớp đã bắt đầu và ta dừng tiến trình luyện. Chú ý rằng, nếu sai số kiểm tra không hề tăng lên, tức là mạng không có đủ số nút ẩn để quá khớp. Khi đó mạng sẽ không có đủ số nút cần thiết để thực hiện tốt nhất. Do vậy nếu hiện tượng quá khớp không hề xảy ra thì ta cần bắt đầu lại nhưng sử dụng nhiều nút ẩn hơn.

## 2.5. Đánh giá mô hình phân lớp

Khi xây dựng một mô hình máy học, chúng ta cần một phép đánh giá để xem mô hình sử dụng có hiệu quả không và để so sánh khả năng của các mô hình. Có rất nhiều cách đánh giá một mô hình phân lớp. Tuỳ vào những bài toán khác nhau mà chúng ta sử dụng các phương pháp khác nhau. Trong phần này tôi sẽ đề cập đến một số phương pháp và tiêu chí đánh giá mô hình máy học.

### 2.5.1. Phương pháp Hold-Out

Phương pháp này phù hợp với tập dữ liệu lớn.

Tập dữ liệu  $D$  được chia thành hai tập con không giao nhau:

- + Tập huấn luyện  $D_{train}$  được sử dụng để huấn luyện hệ thống.

+ Tập kiểm thử  $D_{test}$  được sử dụng để đánh giá hiệu năng của hệ thống đã học.

➔  $D = D_{train} \cup D_{test}$  và thông thường là  $|D_{train}| \gg |D_{test}|$

#### **Một số yêu cầu khi sử dụng phương pháp Hold-out:**

+ Bất kỳ mẫu nào nào thuộc vào tập kiểm thử  $D_{test}$  đều không được sử dụng trong quá trình huấn luyện hệ thống nghĩa là không thuộc vào tập  $D_{train}$

+ Bất kỳ mẫu nào được sử dụng huấn luyện hệ thống nghĩa là thuộc vào tập  $D_{train}$  đều không được sử dụng trong giai đoạn đánh giá hệ thống nghĩa là không thuộc vào tập  $D_{test}$

+ Các mẫu kiểm thử trong tập  $D_{test}$  cho phép một đánh giá không thiên vị đối với hiệu năng của hệ thống.

Các lựa chọn thường gặp:  $|D_{train}| = 2/3 |D|$ ;  $|D_{test}| = 1/3 |D|$

Tuy nhiên có trường hợp mô hình đề xuất cho kết quả tốt với tập Testing Data ( $D_{test}$ ) nhưng áp dụng với tập dữ liệu khác thì kết quả lại không được như ý muốn.

Giả sự mô hình rơi vào trường hợp Overfitting - là hiện tượng mô hình tìm được quá khớp với dữ liệu training ➔ việc chia data làm 2 phần **Training Data** và **Testing Data** thì vẫn chưa thể đưa ra kết luận chính xác cho mô hình được... do đó trong luận văn này, quá trình huấn luyện và kiểm thử mô hình được chia thành 3 giai đoạn:

+ **Giai đoạn 1:** Huấn luyện hệ thống (= Học hàm mục tiêu), Sử dụng Training Data để tìm tham số và tạo ra mô hình.

+ **Giai đoạn 2:** Tối ưu giá trị các tham số của hệ thống. Sử dụng validation data để đánh giá độ chính xác của mô hình. Nếu độ chính xác thấp, tuning parameter để nâng cao độ chính xác của mô hình.

+ **Giai đoạn 3:** Sau cùng tiến hành đánh giá độ chính xác với Testing data.

Điều này có nghĩa là ngoài 2 tập huấn luyện ( $D_{train}$ ) và tập kiểm tra ( $D_{test}$ ) ta cần có 1 tập chứa các mẫu để điều chỉnh tối ưu các giá trị tham số trong quá trình huấn luyện hệ thống ( $D_{validation}$ ). Tập kiểm thử ( $D_{test}$ ) không được sử dụng cho mục đích tối ưu (điều chỉnh) tham số có nghĩa là các mẫu trong tập kiểm thử không thể được sử dụng (theo bất kỳ cách nào) trong tập  $D_{validation}$

Đối với phương pháp này, dựa trên tập dữ liệu kiểm thử ( $D_{test}$ ) ta đánh giá được độ chính xác của mô hình thông qua tỷ lệ phần trăm lượng dữ liệu được phân loại đúng mà không chỉ ra được cụ thể mỗi loại được phân loại như thế nào, lớp nào được phân loại đúng nhiều nhất, và dữ liệu thuộc lớp nào thường bị phân loại nhầm vào lớp khác.

### 2.5.2. Phương pháp True/False Positive/Negative

Phương pháp đánh giá này thường được áp dụng cho các bài toán phân lớp có hai lớp dữ liệu. Cụ thể hơn, trong hai lớp dữ liệu này có một lớp **nghiêm trọng hơn** lớp kia và cần được dự đoán chính xác.

**Ví dụ:** trong bài toán phát hiện bất thường đậm độ trên hình ảnh X-Quang phổi của luận văn này có 2 lớp (normal và abnormal) và việc phát hiện bất thường (abnormal) là lớp dữ liệu nghiêm trọng hơn.

Trong phương pháp này, ta thường định nghĩa lớp dữ liệu quan trọng hơn cần được xác định đúng là lớp **Positive**, lớp còn lại được gọi là **Negative**. Ta định nghĩa **True Positive (TP)**, **False Positive (FP)**, **True Negative (TN)**, **False Negative (FN)** dựa trên **confusion matrix** chưa chuẩn hoá như sau:

**Bảng 2-3:** Định nghĩa True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) dựa trên confusion matrix chưa chuẩn hoá.

	<b>Dự đoán:</b> Positive	<b>Dự đoán:</b> Negative
<b>Thực tế:</b> Positive	True Positive (TP)	False Negative (FN)
<b>Thực tế:</b> Negative	False Positive (FP)	True Negative (TN)

- + **TP:** Số lượng các mẫu thuộc lớp Positive được phân lớp đúng.
- + **FP:** Số lượng các mẫu không thuộc lớp Positive nhưng lại được dự đoán thuộc lớp Positive.
- + **TN:** Số lượng các mẫu thuộc lớp Negative được phân lớp đúng.
- + **FN:** Số lượng các mẫu không thuộc lớp Negative nhưng lại được dự đoán thuộc lớp Negative.

Ta thường quan tâm đến **TPR**, **FNR**, **FPR**, **TNR**, với **R-Rate** dựa trên *normalized confusion matrix* như sau:

**Bảng 2-4:** Tính toán các giá trị: TPR, FNR, FPR, TNR.

	<b>Dự đoán:</b> Positive	<b>Dự đoán:</b> Negative
<b>Thực tế:</b> Positive	$TPR = TP / (TP + FN)$	$FNR = FN / (TP + FN)$
<b>Thực tế:</b> Negative	$FPR = FP / (FP + TN)$	$TNR = TN / (FP + TN)$

+ **TPR:** Tỷ lệ nhận diện Positive đúng trên tổng số các trường hợp Positive. Tỷ lệ này nhằm **đánh giá độ nhạy** của mô hình đề xuất.

$$TPR = \frac{TP}{TP + FN} = \frac{TP}{\sum \text{Positive sample}} \quad (2.24)$$

+ **FPR:** Tỷ lệ nhận diện nhầm mẫu Negative thuộc về lớp Positive hay còn gọi là tỷ lệ báo động nhầm (False Alarm Rate):

$$FPR = \frac{FP}{FP + TN} = 1 - TNR \quad (2.25)$$

+ **TNR:** Tỷ lệ nhận diện Negative đúng trong tổng số các trường hợp Negative. Tỷ lệ này nhằm **đánh giá độ đặc hiệu** của mô hình.

$$TNR = \frac{TN}{TN + FP} = \frac{TN}{\sum \text{Negative sample}} \quad (2.26)$$

+ **FNR:** Tỷ lệ nhận diện nhầm mẫu Positive thuộc về lớp Negative hay còn gọi là tỷ lệ bỏ sót (Miss Detection Rate):

$$FNR = \frac{FN}{TP + FN} = 1 - TPR \quad (2.27)$$

**Với phương pháp này thì độ chính xác của mô hình theo từng trường hợp cụ thể như sau:**

+ Độ chính xác tổng quát: Tỷ lệ của tất cả trường hợp phân loại Đúng (không phân biệt negative/positive) trên toàn bộ trường hợp trong mẫu kiểm thử Testing data set.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.28)$$

+ Khi 2 nhãn Positive/Negative bị mất cân đối, một tiêu chí khác phù hợp hơn là Balanced accuracy (BAC) hay độ chính xác sau cân bằng:

$$\text{BAC} = \frac{TPR + TNR}{2} \quad (2.29)$$

+ Đối với tính toán độ chính xác cho những mục tiêu chuyên biệt hơn như là: tỷ lệ thực sự thuộc lớp positive trên tổng số các trường hợp được gán nhãn “Positive” (PPV) hay nói cách khác là khả năng phân loại Positive chính xác của mô hình.

$$PPV = \frac{TP}{TP + FP} \quad (2.30)$$

hoặc ngược lại là với mục tiêu đo lường khả năng loại trừ chính xác có nghĩa là khả năng phân loại Negative chính xác của mô hình (NPV):

$$NPV = \frac{TN}{TN + FN} \quad (2.31)$$

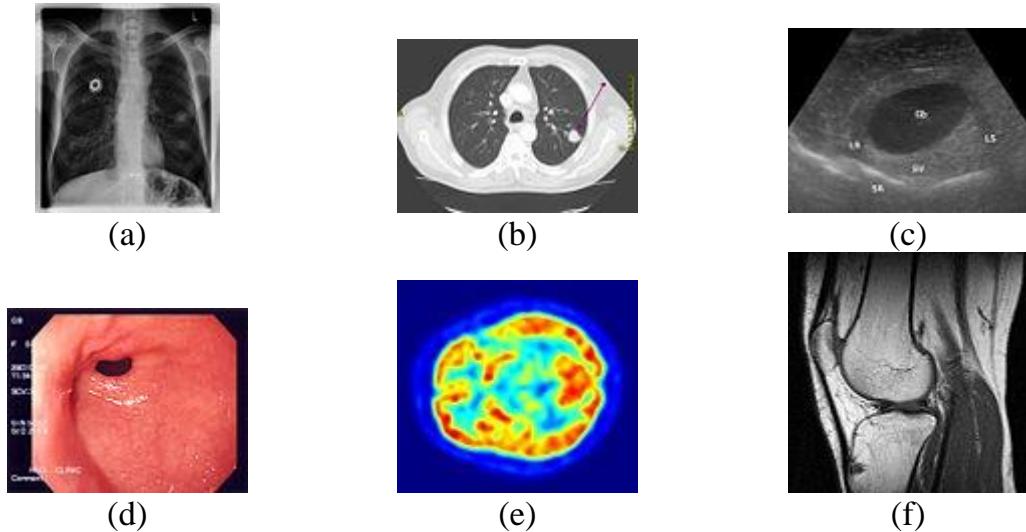
## **2.6. *Ảnh y khoa.***

### **2.6.1. *Giới thiệu:***

Trong chẩn đoán cận lâm sàng trong y khoa thì chẩn đoán dựa trên hình ảnh thu được từ các thiết bị, máy y tế (*chẩn đoán hình ảnh*) ngày càng chiếm một vai trò quan trọng, nhất là ngày nay với sự trợ giúp của các thiết bị, máy y tế hiện đại, công nghệ cao có các phần mềm tin học hỗ trợ khiến cho hình ảnh rõ nét và chính xác hơn.

Ảnh y khoa là sản phẩm của quá trình áp dụng các kỹ thuật thông qua các thiết bị máy móc để tái tạo ra hình ảnh cơ thể con người, hoặc một phần bộ phận cơ thể con người phục vụ cho mục đích chẩn đoán hoặc kiểm tra cận lâm sàng. Ảnh y khoa theo nghĩa rộng thì nó là một phần của hình ảnh sinh học và kết hợp X - quang, nội soi, siêu âm, CT (Computed Tomography), ... dùng trong chẩn đoán điều trị bệnh lý của con người.

Một số hình ảnh y khoa:



Hình 2-28: Một số hình ảnh y khoa [15]

(a) - Ảnh X quang ngực, (b) - Ảnh CT scan ngực, cắt ngang, (c) - Ảnh siêu âm túi mật, (d) - Ảnh nội soi dạ dày, (e) - Ảnh PET não, cắt ngang, (f) - Ảnh MRI đầu gối, đọc mặt cắt ngang

## 2.6.2. *Hình ảnh X-Quang*

### 2.6.2.1. *Sự ghi hình X-Quang*

Sự ghi hình X quang (chụp X - quang) của các bộ phận thăm khám được thực hiện trên phim hoặc giấy ảnh. Để ghi được hình trên phim X quang thì tia X phải được phát xạ với một điện thế cao (từ 50KV đến 100 hoặc 150 KV) và với cường độ dòng qua bóng X quang lớn (từ 100-200mA, và các máy hiện đại hiện nay có thể lên tới 500 đến 1000KV). Hai yếu tố này nhằm đảm bảo cho sự ghi hình nhanh tránh được hình nhiễu của các cơ quan động (như tim, ống tiêu hoá.v.v.) và phù hợp với thời gian nín thở của bệnh nhân. Phim X quang có cấu tạo cơ bản là 2 mặt được tráng bởi nhũ tương muối bạc (bromua bạc). Phim được ép vào giữa 2 tấm tăng quang đặt trong cassette. Bên mặt tấm tăng quang được phủ bằng một lớp chất phát huỳnh quang (thường là Tungstat cadmi). Dưới tác dụng của tia X các lớp huỳnh quang này sẽ phát quang và tác dụng lên phim để ghi hình bộ phận thăm khám mà nó truyền qua. Tia X chỉ tác dụng lên phim khoảng 10% còn lại khoảng 90% tác dụng này là do ánh sáng

huỳnh quang phát ra từ tám tảng quang. Chụp X quang không chuẩn bị bao gồm các kỹ thuật chụp X quang các bộ phận của cơ thể như chụp xương khớp, chụp bụng, chụp sọ não, chụp cột sống, chụp phổi, chụp hệ tiết niệu.v.v. không sử dụng được chất cản quang. Chụp X quang có chuẩn bị được dùng để chỉ những kỹ thuật X quang quy ước có sử dụng được chất cản quang (Barysulfat, các thuốc cản quang tiêm tĩnh mạch) như chụp lưu thông thực quản-dạ dày-tá tràng với baryt, chụp niệu đồ tĩnh mạch, ...

#### **2.6.2.2. Chất lượng hình ảnh X-Quang**

Chất lượng hình ảnh Y học nói chung, phụ thuộc vào nhiều yếu tố như phương pháp ghi hình, đặc điểm của thiết bị, người vận hành kỹ thuật... Chất lượng hình ảnh phụ thuộc vào 5 yếu tố sau: độ tương phản, hình mờ, hình nhiễu (noise), ảnh giả (artifact), hình biến dạng. Riêng hình ảnh X quang có hai yếu tố quan trọng:

- Độ tương phản (contrast): tức là độ đối quang, do sự khác nhau giữa vùng sáng và vùng đen trên phim, nó phụ thuộc vào kỹ thuật chụp và người quan sát. Ta biết rằng sau khi đi xuyên qua cơ thể tia X bị hấp thụ và bị suy giảm một cách khác nhau, vì vậy sự tác dụng lên nhữnг tương trên phim cũng khác nhau: nơi nào không bị suy giảm sẽ tạo nên vùng đen, nơi nào bị suy giảm nhiều tạo nên vùng trắng. Tương tự ta có những vùng xám nhiều hay ít tùy thuộc vào hệ số hấp thụ, và xuất hiện thuật ngữ “nấc thang xám” (gray scale). Trong hình ảnh X - quang thường quy có 4 nấc cơ bản là đen của không khí, xám sẫm của mỡ, xám nhạt của nước và mỏnг, trắng của xương.

- Sự rõ nét của hình ảnh (sharpness): là sự phân biệt giữa các đường khác nhau trên phim, độ rõ nét càng cao đường bờ càng rõ. Nó phụ thuộc vào nhiều yếu tố khác nhau như:

- + Kích thước tiêu điểm phát tia, nếu càng nhỏ độ nét càng cao và ngược lại.
- + Khoảng cách vật phim, càng gần phim càng rõ và ngược lại.
- + Sự cố định của vật tạo cho hình ảnh rõ nét, nếu chuyển động tạo hình mờ.
- + Thời gian chụp, càng ngắn hạn chế được sự chuyển động, hình ảnh càng rõ.

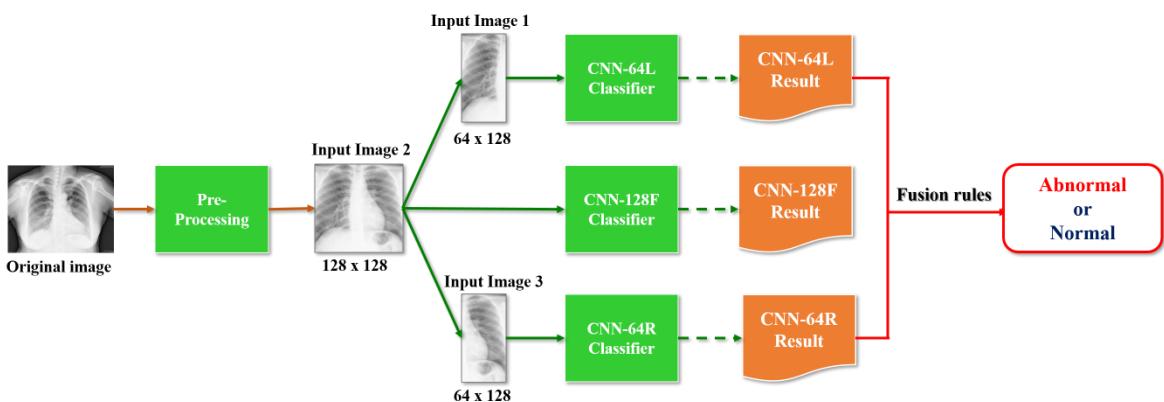
## CHƯƠNG 3 - XÂY DỰNG MÔ HÌNH HỌC SÂU PHÁT HIỆN BẤT THƯỜNG ĐẬM ĐỘ TRÊN HÌNH ẢNH X- QUANG PHỔI

Trong phần này, học viên sẽ trình bày về mô hình Multi-CNNs sử dụng để phát hiện bất thường đậm độ trên hình ảnh X-quang phổi.

### 3.1. Kiến trúc mô hình Multi-CNNs

#### 3.1.1. Giới thiệu

Mô hình được mô tả như Hình 3-1:



Hình 3-1: Mô tả mô hình học sâu Multi-CNNs

Mô hình Multi-CNNs bao gồm 3 thành phần chính, mỗi thành phần là một kiến trúc mạng CNN tương ứng với 3 thành phần của hình ảnh X-Quang phổi, mỗi thành phần được gọi lần lượt như sau:

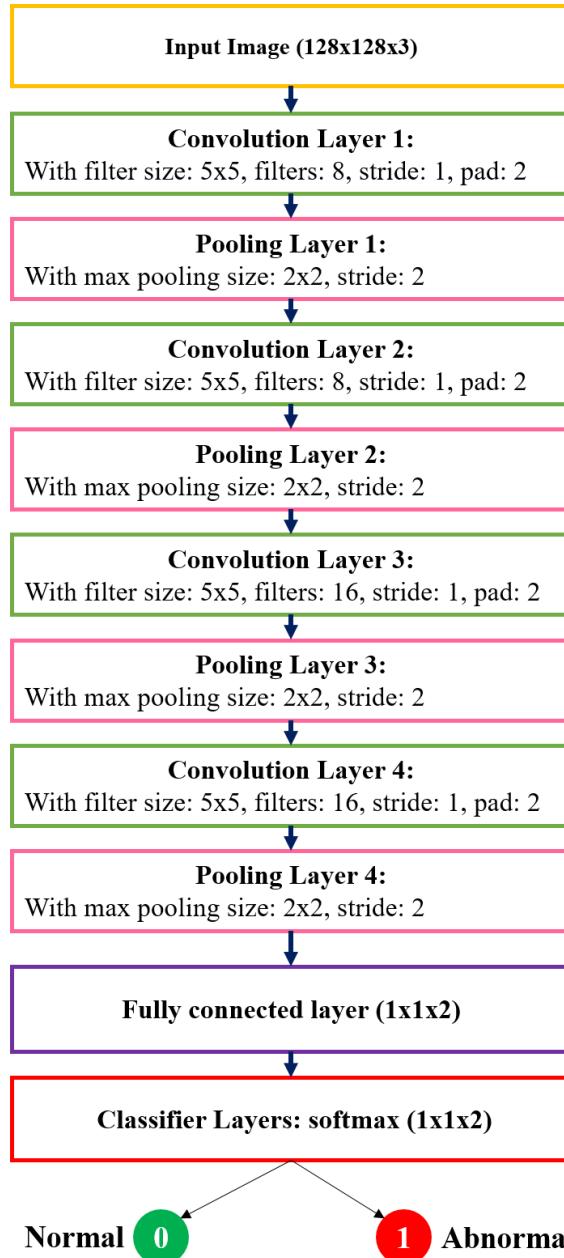
- + CNN-64L: Kiến trúc mạng CNN sử dụng để phân lớp hình ảnh phổi trái.
- + CNN-128F: Kiến trúc mạng CNN sử dụng để phân lớp toàn bộ hình ảnh X-quang phổi.
- + CNN-64R: Kiến trúc mạng CNN sử dụng để phân lớp hình ảnh phổi phải.

Với mỗi hình ảnh đầu vào tương ứng với mỗi thành phần CNN của mô hình Multi-CNNs chúng ta sẽ có một giá trị xác suất là ảnh đầu vào thuộc về lớp bình thường (normal) hoặc bất thường (abnormal). Sau đó, áp dụng luật kết hợp - trong luận văn này gọi là Fusion Rule để tính toán và đưa ra kết luận cuối cùng là ảnh thuộc lớp Normal hay Abnormal. Fusion Rule sẽ được học viên trình bày trong phần kế tiếp.

### 3.1.2. Các thành phần CNN của mô hình đề xuất

Như đã giới thiệu ở 3.1.1 mô hình Multi-CNNs có 3 thành phần chính, mỗi thành phần là một kiến trúc mạng CNN. Trong phần này, học viên sẽ trình bày chi tiết các thành phần này.

#### 3.1.2.1. Kiến trúc mạng CNN-128F



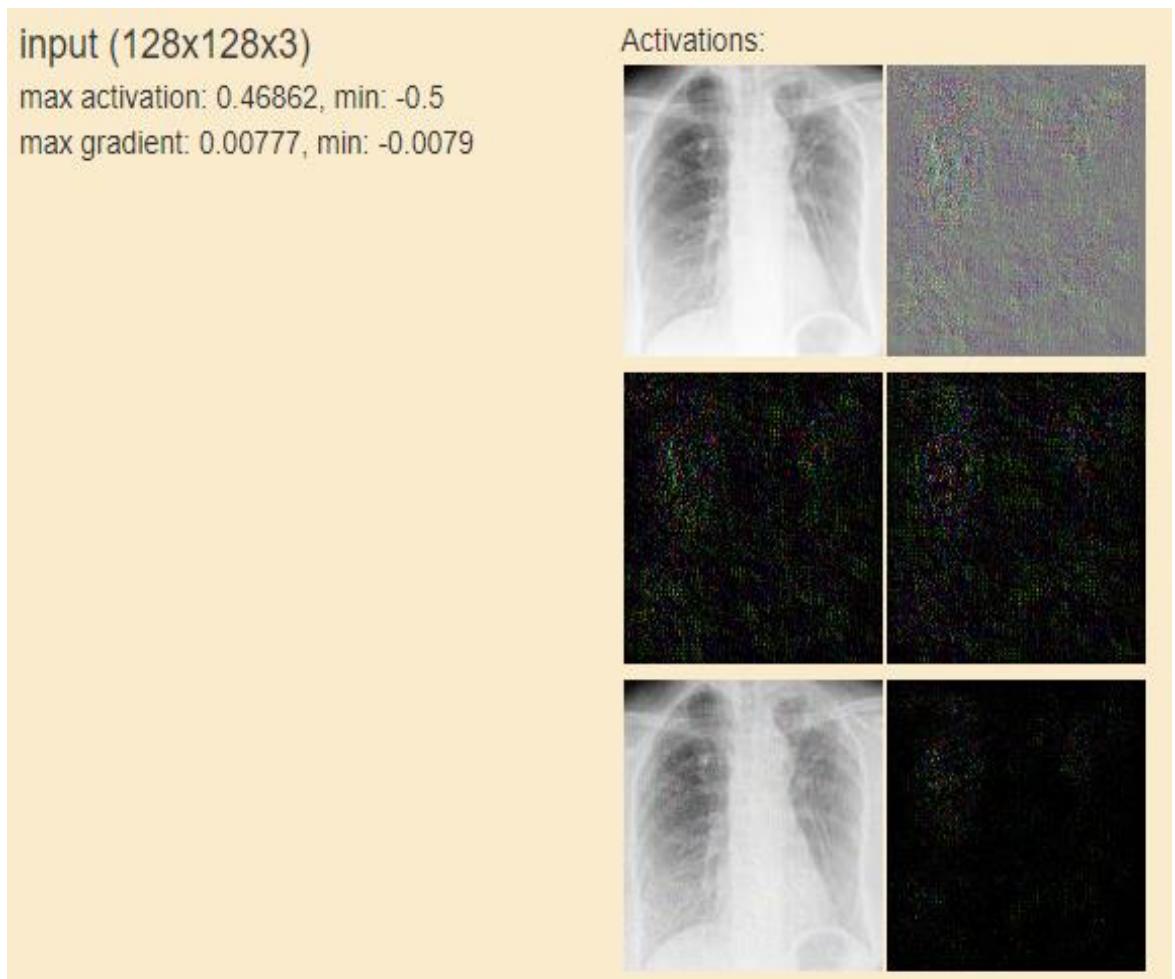
**Hình 3-2:** Kiến trúc mạng của thành phần CNN-128F của mô hình Multi-CNNs

Hình 3-2 mô tả kiến trúc của mạng CNN-128F của mô hình Multi-CNNs. Mạng này được thiết kế gồm 6 lớp, sử dụng 8 bộ lọc với kích thước 5x5 (5x5x8) ở lớp 1 và

2 và 16 bộ lọc với kích thước  $(5 \times 5 \times 16)$  ở lớp 3 và 4. Sử dụng phương pháp Max pooling với kích thước  $2 \times 2$  để giảm kích thước, lớp Fully-Connected (FC) cho ra vector đặc trưng của ảnh, dùng để tính điểm các đối tượng. Và cuối cùng là lớp softmax để đưa ra kết luận cuối cùng là ảnh đầu vào thuộc vào lớp nào.

Quá trình hình ảnh đi qua các tầng trong quá trình huấn luyện được mô tả như sau:

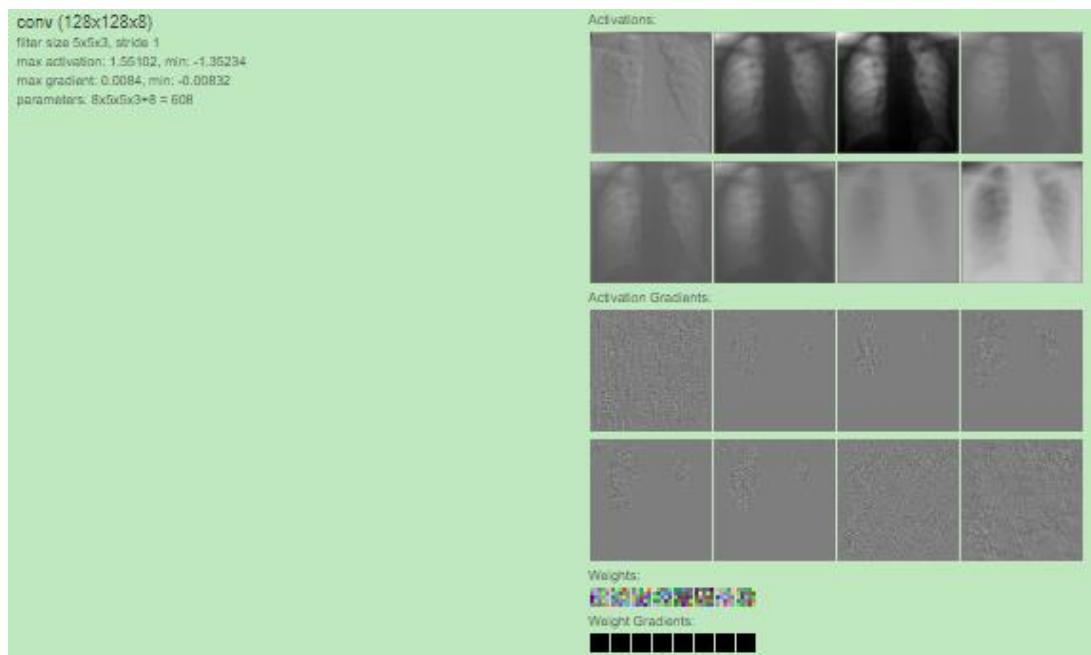
**Tầng Input:** Dữ liệu đầu vào cho quá trình huấn luyện mạng sẽ được định dạng, chuẩn hoá với dữ liệu đầu vào. Ảnh đầu vào có kích thước  $128 \times 128$  và có chiều sâu (depth) = 3. Hình 3-3.



**Hình 3-3:** Mô tả hình ảnh đầu vào được đưa vào và tính toán ở tầng Input của kiến trúc CNN-128F

**Tầng CONV 1:** Hình 3-4, Hình 3-5, Hình 3-6

- **Lớp CONV:** Đây là tầng đặc trưng của phương pháp CNN, thực hiện tính tích chập trên ma trận đầu vào và ma trận mặt nạ.



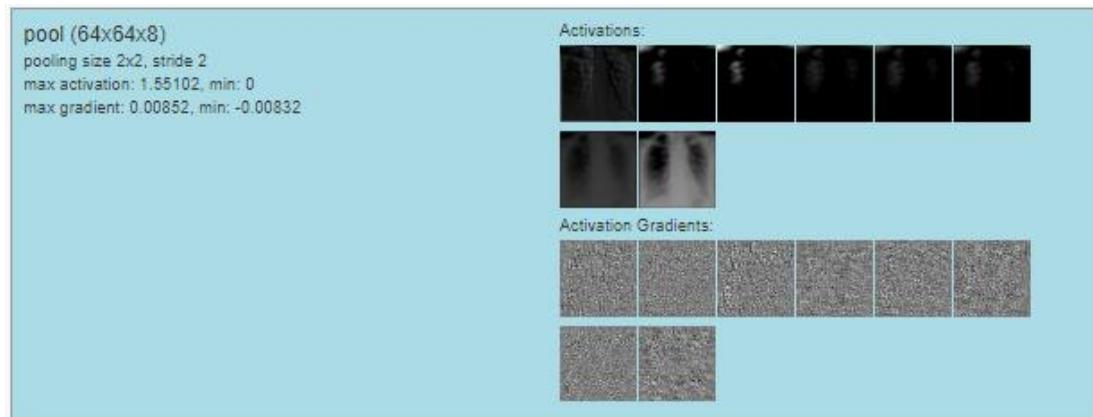
**Hình 3-4:** Mô tả trực quan tầng conv của kiến trúc mạng CNN-128F

- **Lớp ReLU:** Tầng này sử dụng công thức relu (công thức 2.12) trên toàn bộ ma trận là đầu ra của tầng conv.

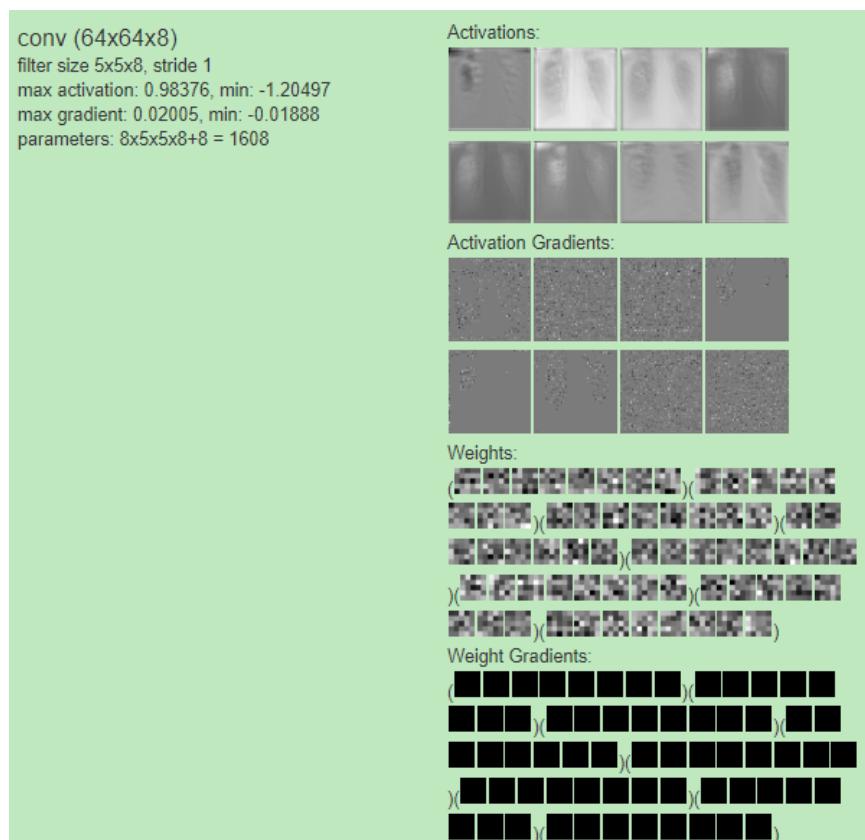


**Hình 3-5:** Mô tả trực quan tầng relu của kiến trúc mạng CNN-128F

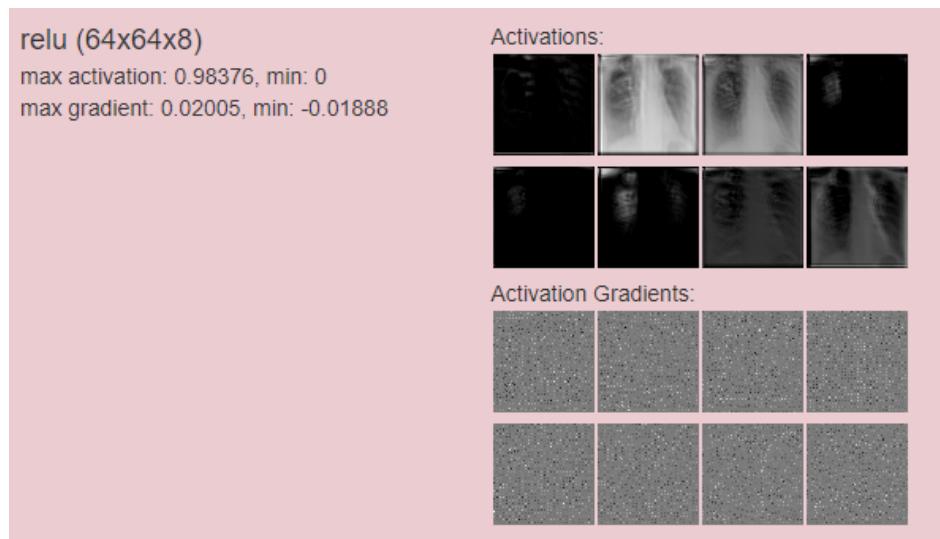
- **Lớp Pool:** tiến hành lấy mẫu con, trong kiến trúc này sử dụng phương pháp Max Pooling tại tầng Pool. Tại tầng này kích thước ảnh đã giảm từ 128 x 128 xuống 64 x 64. Hình 3-6



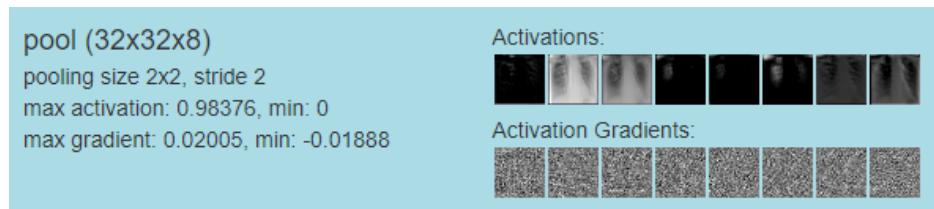
**Hình 3-6:** Mô tả trực quan quá trình xử lý tại tầng Pool của kiến trúc mạng CNN-128F  
Tương tự ở Layer 2 (lớp 2) của kiến trúc mạng CNN-128F được biểu diễn như sau  
**Tầng CONV 2:** Hình 3-7, Hình 3-8, Hình 3-9.



**Hình 3-7:** Mô tả trực quan tầng conv của layer 2 của kiến trúc mạng CNN-128F

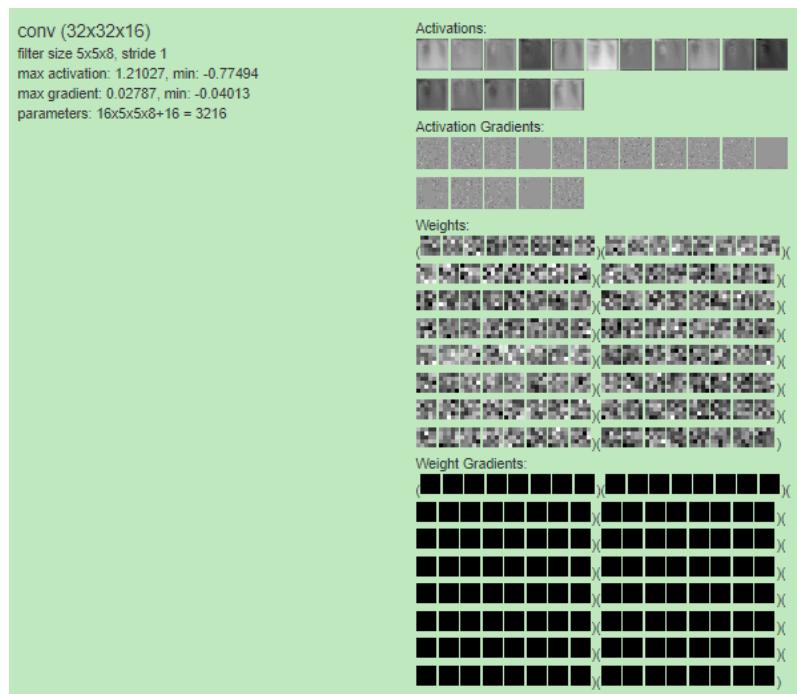


**Hình 3-8:** Mô tả trực quan ReLU của layer 2 của kiến trúc mạng CNN-128F

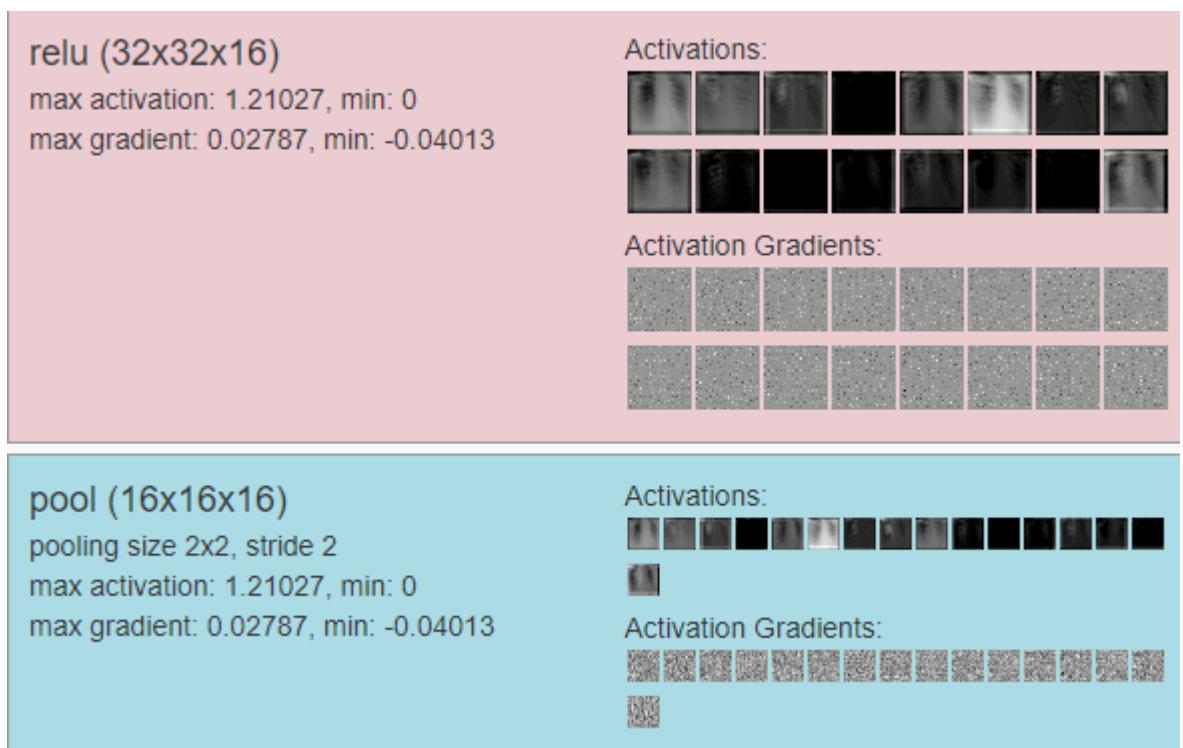


**Hình 3-9:** Mô tả trực tiếp tầng Pool của layer 2 của kiến trúc mạng CNN-128F

**Tầng CONV 3:** Hình 3-10, Hình 3-11



**Hình 3-10:** Mô tả trực quan tầng conv của layer 3 của kiến trúc mạng CNN-128F



**Hình 3-11:** Mô tả trực quan tầng ReLU và Pool của layer 3 của kiến trúc mạng CNN-128F

Tương tự cho Layer 4 của kiến trúc mạng CNN-128F.

**Tầng Fully Connected layer (FC):** Đây là lớp cuối cùng của Kiến trúc mạng CNN-128F, lớp này đóng vai trò phân lớp dữ liệu dựa trên dữ liệu đã được xử lý trước đó.



**Hình 3-12:** Mô tả trực quan lớp FC của kiến trúc mạng CNN-128F

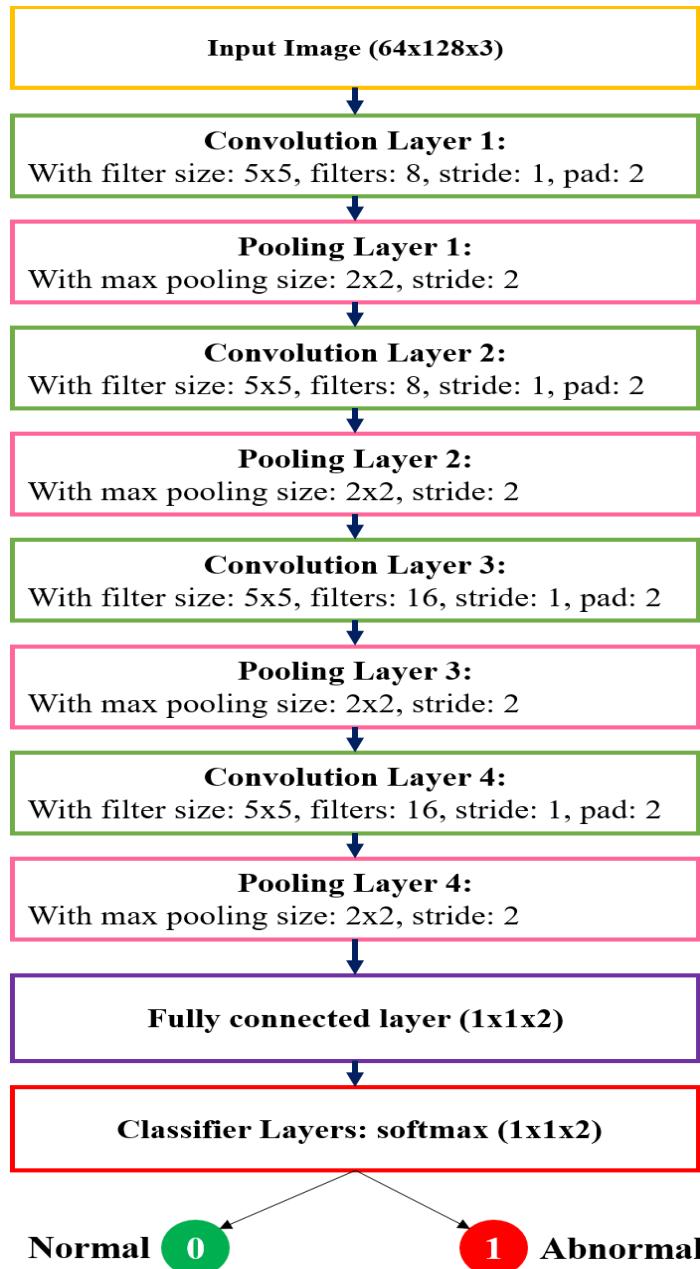
- **Hàm Softmax:** Trong kiến trúc CNN-128F hàm Softmax được sử dụng ở lớp cuối cùng (FC) để phân loại.



**Hình 3-13:** Mô tả trực các tính toán của hàm softmax của kiến trúc mạng CNN-128F

### 3.1.2.2. Kiến trúc mạng CNN-64L và CNN-64R

Trong luận văn này sử dụng kiến trúc mạng như hình 3-13 để huấn luyện cho hình ảnh x-quang phổi trái và x-quang phổi phải. Kiến trúc mạng CNN-64L/R được thiết kế cho hình ảnh đầu vào có kính thước  $64 \times 128$ .



Hình 3-14: Kiến trúc mạng của thành phần CNN-64L và CNN-64R  
của mô hình Multi-CNNs

Tương tự như việc hiển thị trực quan các bước xử lý ảnh đầu vào thông qua các tầng của kiến trúc mạng CNN-128F, trong phần này học viên chỉ đề cập đến tầng input, các tầng trong layer 1 và lớp fully connected.

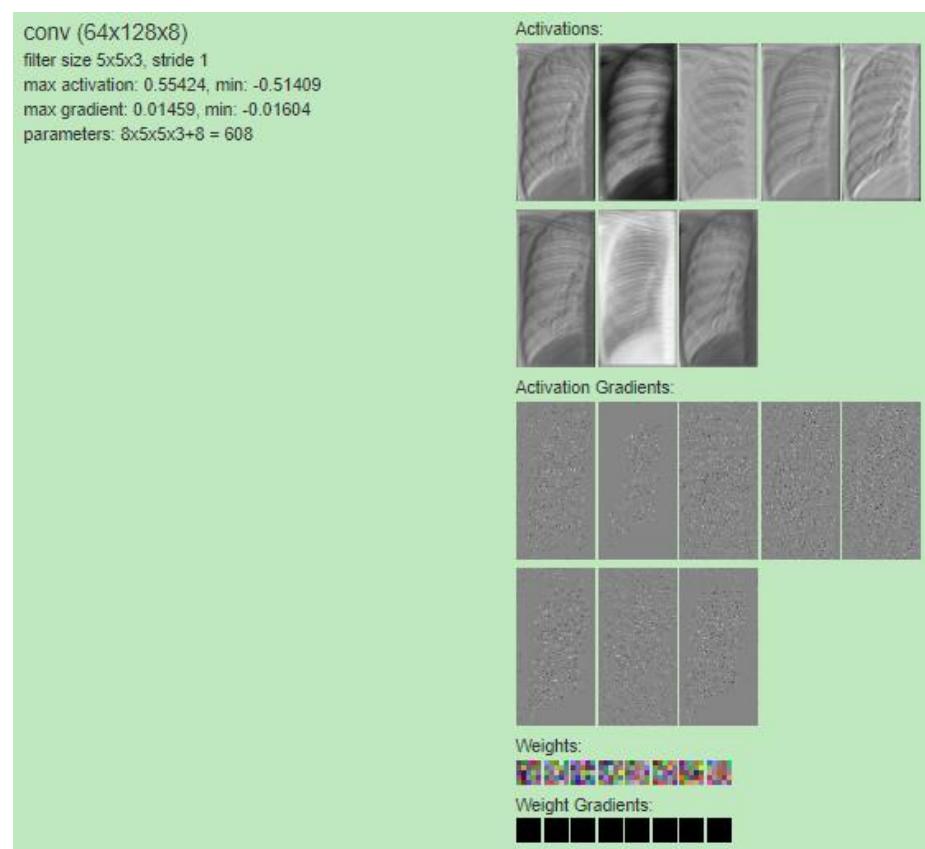
**Tầng INPUT:** Ảnh đầu vào có kích thước  $64 \times 128$  và có chiều sâu (depth) = 3 được mô tả trực quan như hình 3-15.



**Hình 3-16:** Mô tả hình ảnh đầu vào được đưa vào và tính toán ở tầng Input của kiến trúc CNN-64L/R

### Tầng CONV 1:

#### - Lớp CONV:



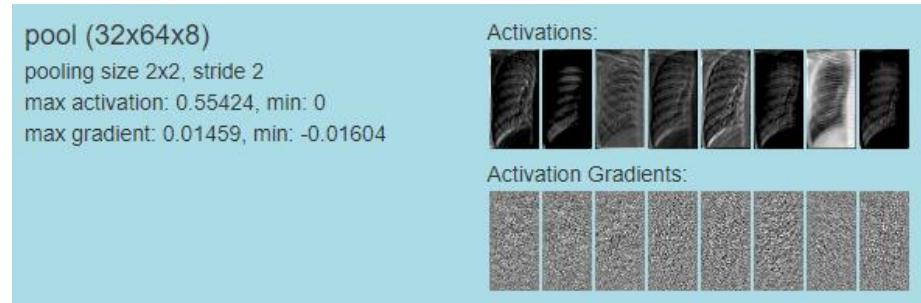
**Hình 3-17:** Mô tả trực quan tầng conv của kiến trúc mạng CNN-64L/R

### - Lớp ReLU:



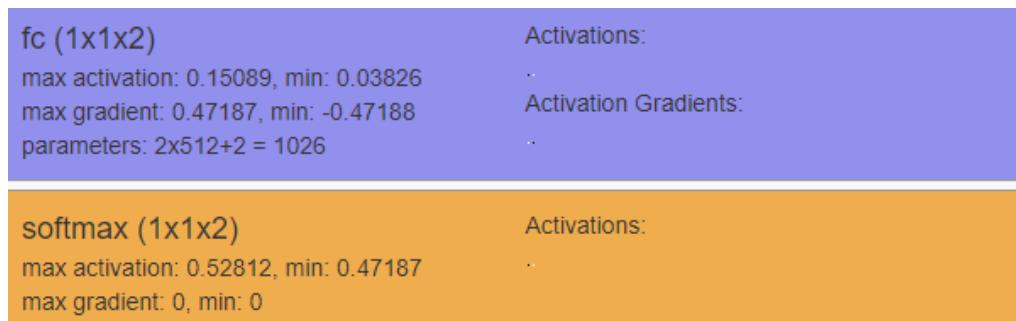
Hình 3-18: Mô tả trực quan tầng ReLU của kiến trúc mạng CNN-64L/R

### - Lớp Pooling:



Hình 3-19: Mô tả trực quan tầng Pool của kiến trúc mạng CNN-64L/R

### Tầng Fully Connected layer (FC) và hàm Softmax:



Hình 3-20: Mô tả trực quan lớp FC và hàm Softmax của kiến trúc mạng CNN-64L/R

### 3.2. Các quy tắc kết hợp (fusion rules) dựa trên tập các trường hợp xảy ra

Như đã đề cập ở trên, trong kiến trúc của mô hình multi-CNNs, ta phải sử dụng một quy tắc kết hợp để đưa ra kết luận cho ảnh đầu vào là thuộc lớp nào dựa trên 3 kết quả từ 3 thành phần CNN của mô hình multi-CNNs.

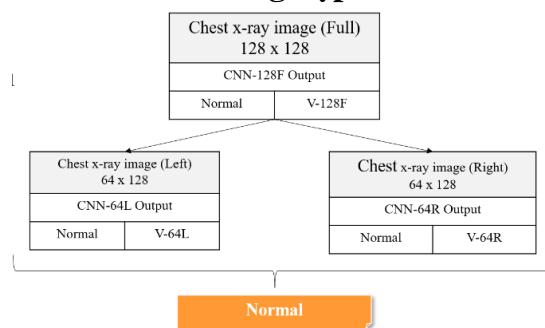
Đối với mỗi đầu ra của một thành phần CNN trong mô hình, ta lấy được một kết quả gồm có 2 giá trị: ảnh đầu vào thuộc vào lớp bất thường (Abnormal) hay bình thường (Normal) cùng với giá trị xác xuất thuộc vào mỗi lớp của hình ảnh đó. Các giá trị xác xuất được định nghĩa trong luận văn này như sau:

- + **V128F**: Giá trị xác xuất sau khi tính toán cho hình ảnh đầu vào với CNN-128F.
- + **V64L**: Giá trị xác xuất sau khi tính toán cho hình ảnh đầu vào với CNN-64L.
- + **V64R**: Giá trị xác xuất sau khi tính toán cho hình ảnh đầu vào với CNN-64R.

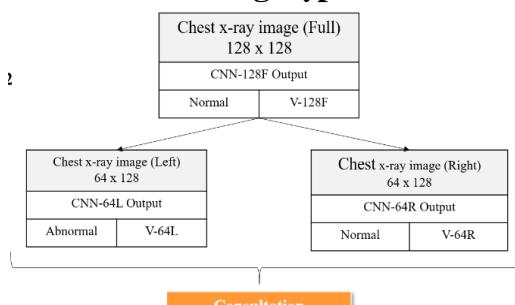
#### 3.2.1. Tập các trường hợp xảy ra của mô hình phân lớp

Với kết quả phân lớp mỗi bức ảnh đầu vào của mô hình, học viên chia ra thành 8 trường hợp có khả năng xảy ra. Kết quả kết hợp ở mỗi trường hợp có thể cho ra 1 trong 3 kết luận sau: Bình thường (Normal), Bất thường (Abnormal) hoặc Hội chẩn (Consultation), cụ thể như sau:

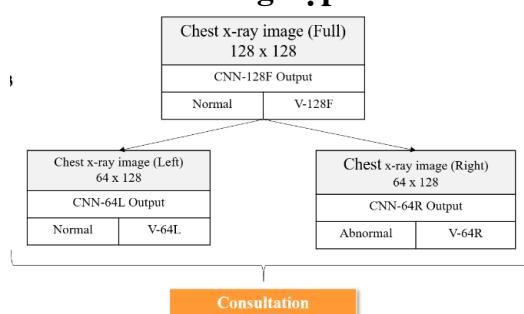
**Trường hợp 1:**



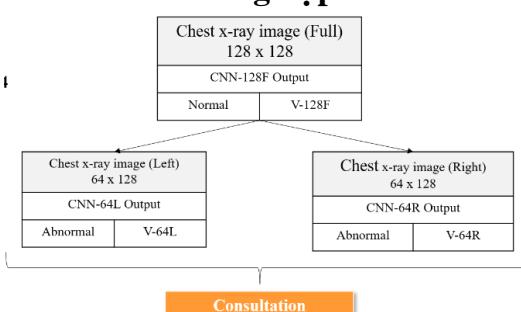
**Trường hợp 2:**



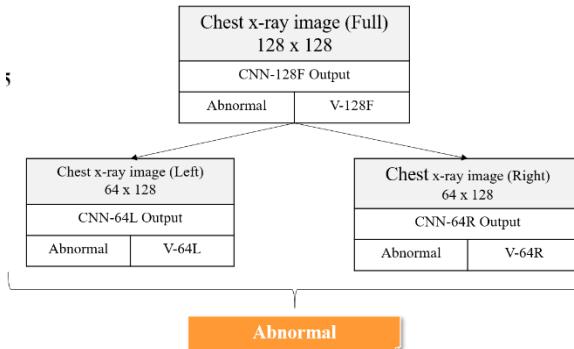
**Trường hợp 3:**



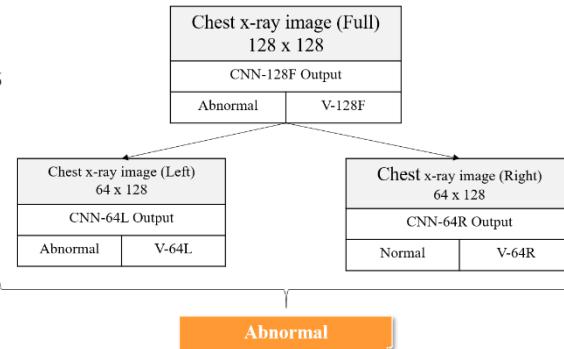
**Trường hợp 4:**



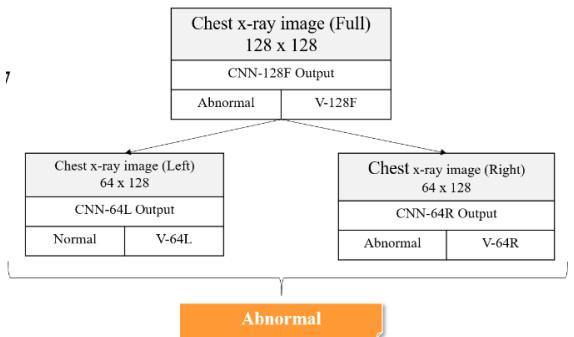
## **Trường hợp 5:**



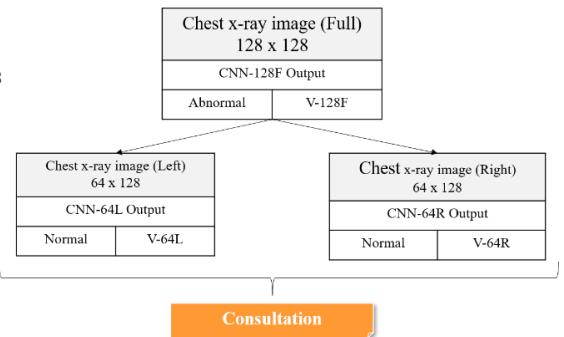
## Trường hợp 6:



## Trường hợp 7:



## Trường hợp 8:



### **3.2.2. Quy tắc kết hợp dựa trên tập các trường hợp xảy ra**

Trong 8 trường hợp trên thì có các trường hợp 2, 3, 4, 8 cho ra kết luận là Hội chẩn (Consultation). Vấn đề đặt ra là phải tìm phương pháp để tính toán giá trị xác xuất để có được kết luận cuối cùng là ảnh đầu vào là Bất thường (Normal) hay Bình thường (Abnormal). Trong luận văn này đề xuất và thực nghiệm với 3 quy tắc sau:

- FR1: Quy tắc kết hợp dựa trên việc tính trung bình giá trị xác xuất.

Trường hợp	Giá trị xác xuất	Kết luận
2	$V = \frac{V128F + V64R + (1 - V64L)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
3	$V = \frac{V128F + V64L + (1 - V64R)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
4	$V = \frac{V128F + (1 - V64L) + (1 - V64R)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
8		$V > 0.5 \rightarrow$ Abnormal $V \leq 0.5 \rightarrow$ Normal

**- FR2: Quy tắc kết hợp dựa trên tổng thẻ.**

Trong quy tắc này xem giá trị xác xuất V128F là gấp đôi so với 2 giá trị còn lại

Trường hợp	Giá trị xác xuất	Kết luận
2	$V = \frac{V128F + V64R + (1 - V64L)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
3	$V = \frac{V128F + V64L + (1 - V64R)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
4	$V = \frac{V128F + (1 - V64L) + (1 - V64R)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
8		$V > 0.5 \rightarrow$ Abnormal $V \leq 0.5 \rightarrow$ Normal

**- FR3: Quy tắc kết hợp dựa trên các vùng xung đột.**

Trong phương pháp này ta chỉ tính trung bình của toàn bộ ảnh và phần (trái hoặc phải) có kết luận ngược với kết quả của CNN-128F.

Trường hợp	Giá trị xác xuất	Kết luận
2	$V = \frac{V128F + (1 - V64L)}{2}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
3	$V = \frac{V128F + (1 - V64R)}{2}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
4	$V = \frac{2 * V128F + (1 - V64L) + (1 - V64R)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
8		$V > 0.5 \rightarrow$ Abnormal $V \leq 0.5 \rightarrow$ Normal

## CHƯƠNG 4 - THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

### 4.1. Dữ liệu thực nghiệm

#### 4.1.1. Bộ cơ sở dữ liệu hình ảnh x-quang ngực

Thực nghiệm là một phần không thể thiếu trong bất cứ việc đánh giá một hệ thống nào đó, nhằm để có thể đánh giá được sự hiệu quả, thích hợp của hệ thống với một vấn đề đặt ra cụ thể. Để quá trình thực nghiệm có thể đánh giá chính xác được hệ thống thì bộ dữ liệu tham gia vào việc đánh giá cũng không kém phần quan trọng. Thông thường để thực nghiệm với các hệ thống máy học, bộ dữ liệu tham gia vào việc đánh giá được chia làm 2 phần chính để thực nghiệm đó là bộ dữ liệu huấn luyện (train) và bộ dữ liệu kiểm thử (test). Cơ sở dữ liệu hình ảnh được sử dụng trong luận văn này gồm 2 bộ dữ liệu, trong đó có 1 bộ dữ liệu được thu thập tại bệnh viện An Bình, TP. Hồ Chí Minh và 1 bộ dữ liệu được thu thập từ thư viện y khoa quốc gia Hoa kỳ [19] Chi tiết từng bộ ảnh được mô tả như bảng 4-1

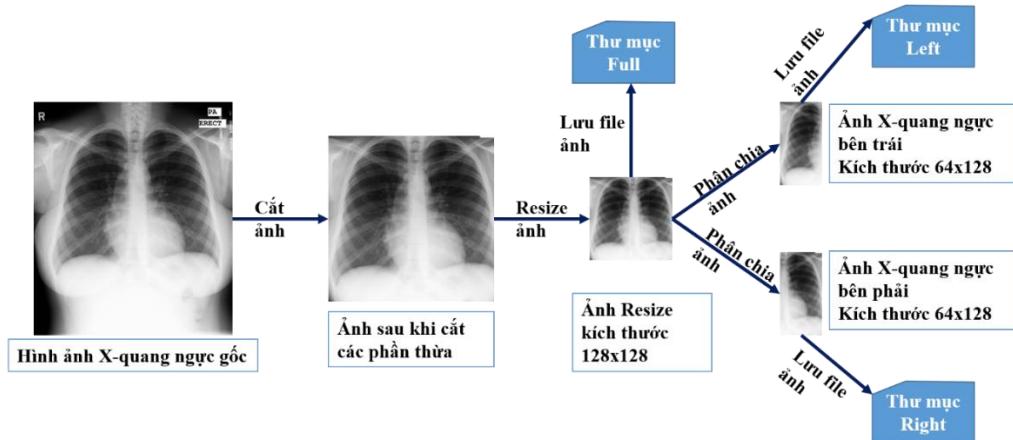
**Bảng 4-1:** Mô tả chi tiết số lượng hình ảnh của từng bộ ảnh

Bộ dữ liệu	Số lượng Normal	Số lượng Abnormal	Tổng
An Bình hospital - Chest X-ray Database (AB-Chest X-rays)	225	225	450
The Shenzhen set - Chest X-ray Database (SZ-Chest X-rays)	326	336	662

Trong luận văn này, học viên tiến hành sử dụng 1 bộ dữ liệu hình ảnh AB-Chest X-rays để tiến hành huấn luyện (traing) và kiểm tra (testing) cho các CNN thành phần của mô hình Multi-CNNs. Sau đó sẽ sử dụng dữ liệu hình ảnh của bộ dữ liệu SZ-Chest X-rays để kiểm tranh đánh giá hiệu năng của mô hình Multi-CNNs.

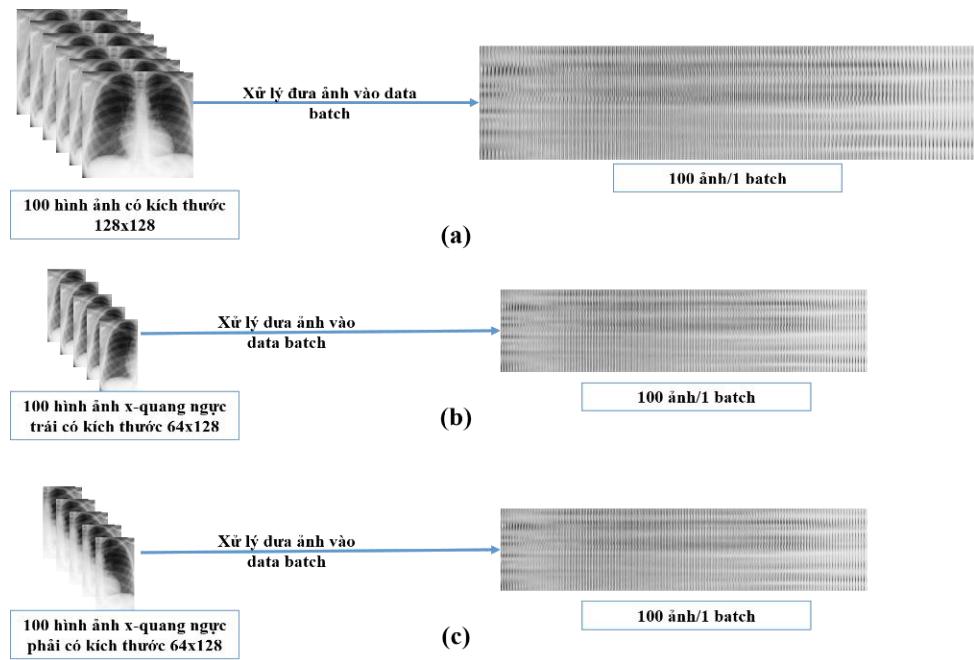
#### 4.1.2. Tiết xử lý hình ảnh và chuẩn hóa dữ liệu huấn luyện và kiểm tra

Với mỗi hình ảnh gốc ban đầu, tiến hành thực hiện các bước cắt, thay đổi kích thước (resize), phân chia (split) và lưu trữ như Hình 4-1.



**Hình 4-1:** Các bước thực hiện tiền xử lý ảnh đầu vào

Sau bước tiền xử lý hình ảnh đầu vào, tiến hành nén các hình ảnh vào một batch để tiến hành đưa vào hệ thống CNN để huấn luyện và kiểm tra. Mỗi batch gồm 100 hình ảnh (*50 hình ảnh bất thường và 50 hình ảnh bình thường*).



**Hình 4-2:** Mô tả quá trình ảnh vào batch dữ liệu để đưa vào hệ thống CNN

(a)-Quá trình đưa ảnh kích thước 128x128 vào batch dữ liệu; (b) và (c)-Quá trình đưa ảnh kích thước 64x128 vào batch dữ liệu

## **4.2. Quá trình thực nghiệm đánh giá mô hình đề xuất**

### **4.2.1. Môi trường thực nghiệm**

#### **4.2.1.1. Nền tảng Phần cứng**

Học viên tiến hành thực nghiệm trên máy vi tính (PC) với cấu hình như sau:

- + Processor: Intel(R) Core (TM) i7-7700 CPU @ 3.60GHz (8 CPUs), ~3.6GHz
- + Memory: 8192MB RAM
- + Disk: INTEL SSD SC2KW360H6
- + VGA card: Không sử dụng (Không có GPU).

#### **4.2.1.2. Công cụ và nền tảng phần mềm sử dụng**

Để xây dựng ứng dụng để kiểm tra đánh giá mô hình đã đề xuất và tiền xử lý hình ảnh học viên đã sử dụng các công cụ phần mềm và các ngôn ngữ lập trình như sau:

- + Hệ điều hành: Microsoft Windows 10 Professional 64bit (Build 17134).
- + Trình duyệt web: Google Chrome.
- + Ngôn ngữ lập trình: HTML&CSS Javascript, Python.
- + Thư viện: ConvNetJS [20], Bootstrap, Jquery
- + Công cụ phần mềm: JPEGCrops0.7.5b, TileMage Image Splitter, Microsoft Excel 2016, XAMPP package.

Công cụ để xây dựng ứng dụng thực nghiệm cho mô hình đó chính là thư viện Javascript ConvNetJS. Thư viện ConvNetJS được đề xuất và xây dựng bởi Karpathy [20] và được mở rộng bởi những đóng góp từ cộng đồng mạng, đây là thư viện hiện thực hóa mô hình ConvNet do Yan Lecun đề xuất. Thư viện bao gồm một số tính năng cơ bản như sau:

- + Các mô-đun mạng nơron chung (các lớp được kết nối hoàn toàn, phi tuyến tính)
- + Chức năng phân loại (SVM / Softmax) và Regression (L2)
- + Khả năng xác định và đào tạo các mạng liên kết để xử lý hình ảnh.
- + Mô-đun Học tập tăng cường thử nghiệm, dựa trên Deep Q Learning.

Điểm đặc biệt của thư viện này là có khả năng thực thi trên trình duyệt web, không yêu cầu nền tảng phần cứng phải có GPU như các thư viện hay nền tảng khác khi xử lý công việc với mô hình CNN hay các mạng học sâu và không yêu cầu phải

có trình biên dịch cũng như phải cài đặt lên máy vi tính. Đây là một điểm hết sức mạnh mẽ của thư viện này, thông qua đó các nhà nghiên cứu có thể tận dụng khả năng của một máy cá nhân và cho chạy ở mọi nơi.

#### **4.2.2. Quá trình huấn luyện các CNN thành phần của mô hình**

Để huấn luyện và kiểm tra cho từng thành phần CNN của mô hình Multi-CNNs, học viên đã chọn lựa 400 hình ảnh từ 1 bộ dữ liệu AB-Chest X-rays phân chia bộ dữ liệu như sau:

**Bảng 4-2:** Mô tả chi tiết số lượng hình ảnh lựa chọn trong bộ dữ liệu để huấn luyện và kiểm tra cho từng thành phần của CNN của mô hình

Bộ dữ liệu	Dữ liệu huấn luyện và đánh giá (Training and Validation dataset)			Dữ liệu kiểm tra (Testing dataset)		
	Normal	Abnormal	Tổng	Normal	Abnormal	Tổng
AB-Chest X-rays	150	150	300	50	50	100

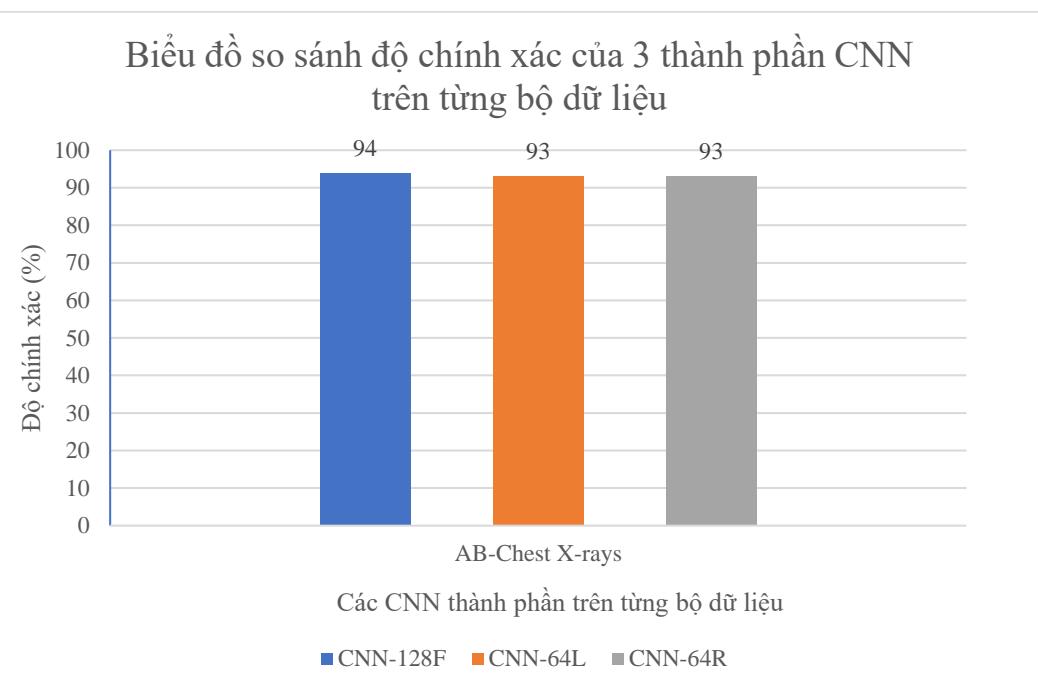
Sau đó tiến hành xử lý dữ liệu hình ảnh theo phương pháp đã nêu tại mục 4.1.2 của luận văn. Để huấn luyện và đánh giá từng CNN thành phần, học viên sử dụng phương pháp Hold-out đã trình bày tại mục 2.4.1 của luận văn và thực hiện như sau:



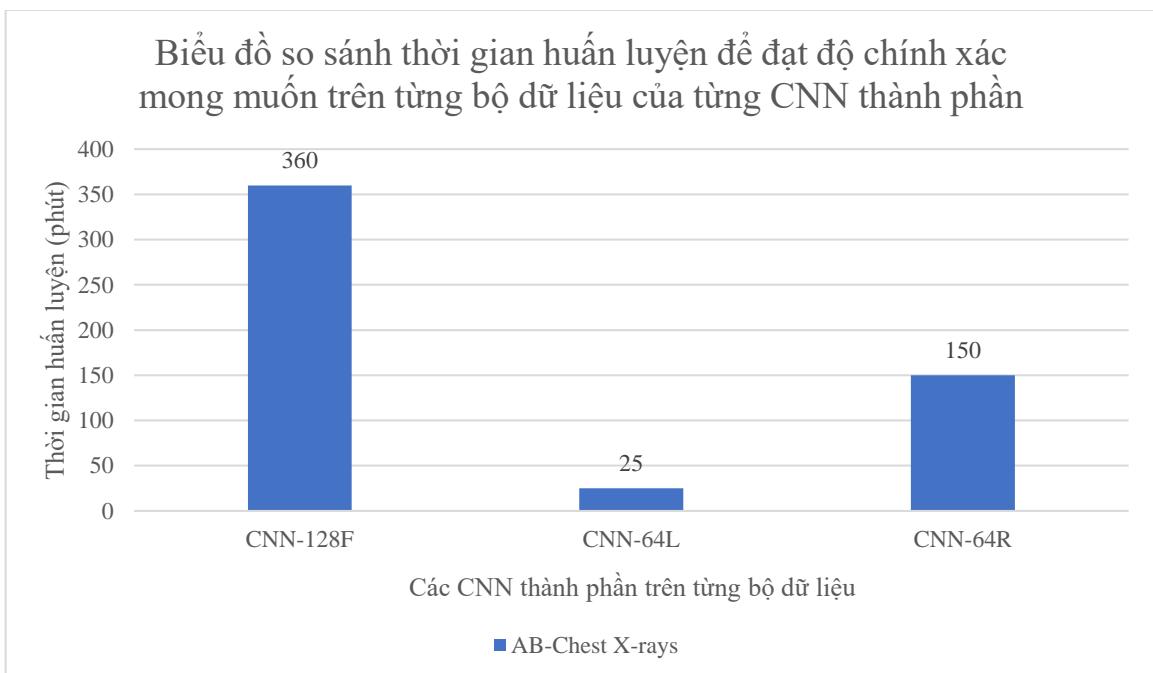
Kết quả huấn luyện của từng thành phần trên từng bộ dữ liệu được mô tả như bảng 4-3.

**Bảng 4-3:** Kết quả huấn luyện các CNN thành phần của mô hình Mult-CNNs

Mạng CNN	CNN-128F		CNN-64L		CNN-64R	
	Độ chính xác	Thời gian	Độ chính xác	Thời gian	Độ chính xác	Thời gian
AB-Chest X-rays	94%	360 phút	93%	25 phút	93%	150 phút



**Hình 4-3:** Biểu đồ so sánh độ chính xác của 3 thành phần CNN



**Hình 4-4:** Biểu đồ so sánh thời gian huấn luyện để đạt được độ chính xác cao nhất của mỗi mạng CNN thành phần

Trong quá trình huấn luyện mỗi thành phần, học viên lưu lại thông tin mạng đạt độ chính xác cao nhất thành file .json để phục vụ cho việc xây dựng và kiểm tra mô hình Multi-CNNs đã đề xuất.

### 4.3. Kết quả thực nghiệm mô hình đề xuất

Để đánh giá độ chính xác của mô hình Multi-CNNs dựa trên quy tắc kết hợp mà luận văn đề xuất, học viên lấy 50 hình ảnh từ bộ dữ liệu AB-Chest X-rays (*50 hình ảnh chưa được sử dụng để Huấn luyện, kiểm tra trên các CNN thành phần*) gồm 25 ảnh bất thường (abnormal) và 25 ảnh bình thường (normal). Kết quả kiểm tra tương ứng với 3 luật kết hợp đã trình bày ở mục 3.2 được thống kê theo bảng 4-4 và 4-5

**Bảng 4-4:** Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu AB-Chest X-rays tương ứng với 3 mạng CNN thành phần

Luật kết hợp	Kết quả kiểm tra trên 50 ảnh của bộ dữ liệu AB-Chest X-rays					
	CNN-128F		CNN-64L		CNN-64R	
	Số lượng	Độ chính xác	Số lượng	Độ chính xác	Số lượng	Độ chính xác
<b>Đúng</b>	46	92%	43	86%	39	78%
<b>Sai</b>	4	8%	7	14%	11	22%

**Bảng 4-5:** Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu AB-Chest X-rays tương ứng với 3 luật kết hợp

Luật kết hợp	Kết quả kiểm tra trên 50 ảnh của bộ dữ liệu AB-Chest X-rays					
	FR1		FR2		FR3	
	Số lượng	Độ chính xác	Số lượng	Độ chính xác	Số lượng	Độ chính xác
<b>Đúng</b>	47	94%	47	94%	48	96%
<b>Sai</b>	3	6%	3	6%	2	4%

#### **4.3.1.2. Kiểm tra mô hình bằng hình ảnh của bộ dữ liệu SZ-Chest X-rays**

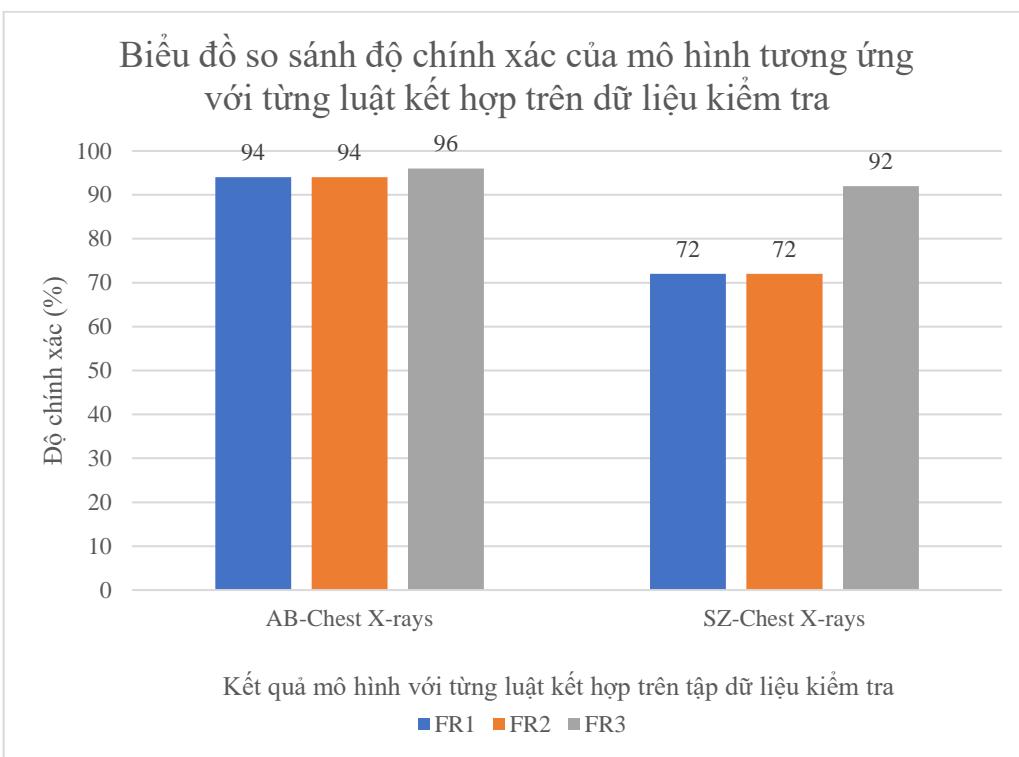
Tiếp tục với kết quả đạt được như mô tả trong bảng 4-4, học viên tiến hành thực nghiệm kiểm tra độ chính xác của mô hình bằng cách sử dụng 150 bức ảnh được lấy từ bộ dữ liệu SZ-Chest X-rays, gồm 75 ảnh bất thường (abnormal) và 75 ảnh bình thường (normal). Kết quả kiểm tra tương ứng với 3 luật kết hợp đã trình bày ở mục 3.2 của luận văn được thống kê theo bảng 4-6, 4-7.

**Bảng 4-6:** Kết quả kiểm tra tên trên 50 hình ảnh thuộc bộ dữ liệu SZ-Chest X-rays tương ứng với 3 mạng CNN thành phần.

Luật kết hợp	Kết quả trên 150 ảnh của bộ dữ liệu SZ-Chest X-rays					
	CNN-128F		CNN-64L		CNN-64R	
	Số lượng	Độ chính xác	Số lượng	Độ chính xác	Số lượng	Độ chính xác
<b>Đúng</b>	131	87.3%	104	69.3%	83	55.3%
<b>Sai</b>	19	12.7%	46	30.7%	67	44.7%

**Bảng 4-7:** Kết quả kiểm tra tên trên 150 hình ảnh thuộc bộ dữ liệu SZ-Chest X-rays tương ứng với 3 luật kết hợp

Luật kết hợp	Kết quả trên 150 ảnh của bộ dữ liệu SZ-Chest X-rays					
	FR1		FR2		FR3	
	Số lượng	Độ chính xác	Số lượng	Độ chính xác	Số lượng	Độ chính xác
<b>Đúng</b>	108	72%	108	72%	138	92%
<b>Sai</b>	42	28%	42	28%	12	8%



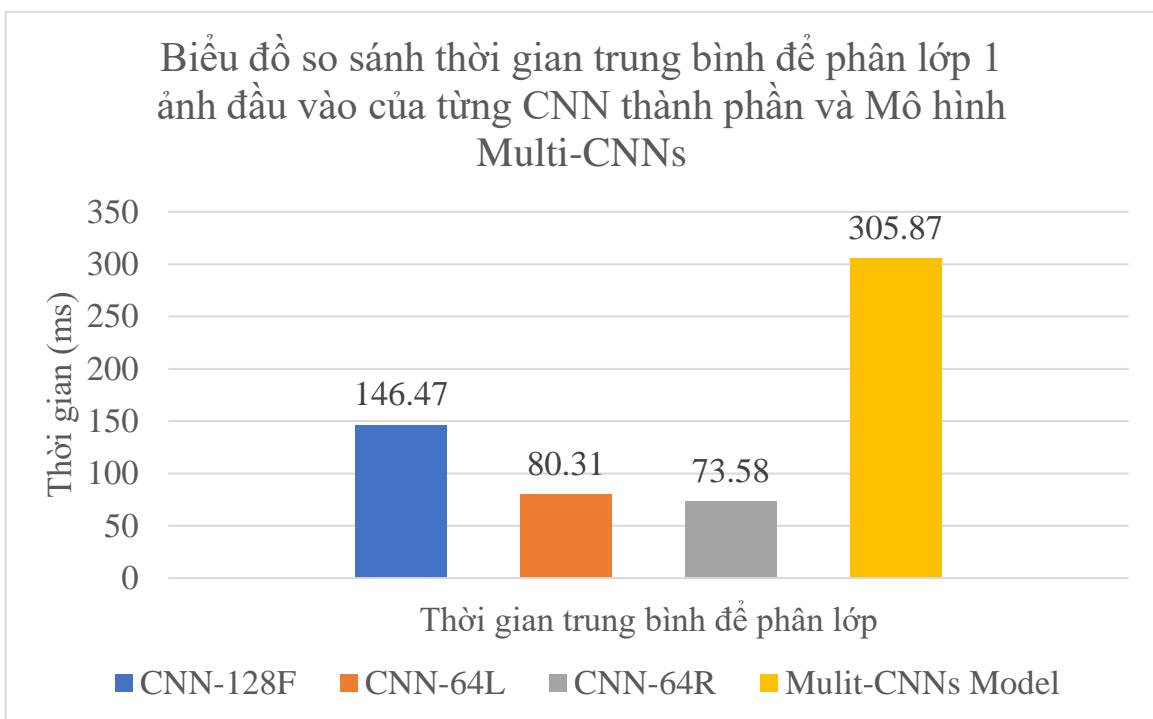
**Hình 4-5:** Biểu đồ so sánh độ chính xác của mô hình trên từng tập dữ liệu ứng với từng phương pháp tính toán kết hợp

Để đánh giá hiệu năng của mô hình đề xuất, học viên cũng thống kê thời gian trung bình để tính toán để đưa ra kết luận cuối cùng cho ảnh đầu vào của từng CNN thành phần và cả mô hình, cụ thể như bảng 4-8.

**Bảng 4-8:** Thời gian trung bình của quá trình tính toán để đưa ra kết luận cuối cùng cho 1 ảnh đầu vào

Mạng	CNN-128F	CNN-64R	CNN-64L	Multi-CNNs
Thời gian				
Thời gian trung bình	146.47 ms	73.58 ms	80.31 ms	305.87 ms

Bảng 4-6 mô tả thời gian trung của quá trình tính toán để đưa ra kết luận cuối cùng cho 1 ảnh đầu vào của từng mạng CNN thành phần và của mô hình Multi-CNNs. Kết quả trong bảng được thống kê bằng cách tính toán việc phân lớp 150 bức ảnh đầu vào.



**Hình 4-6:** Biểu đồ so sánh thời gian trung bình phân lớp 1 ảnh đầu vào của các CNN thành phần và mô hình Multi-CNNs

Trong quá trình thực nghiệm, ngoài việc sử dụng phương pháp Hold-Out để đánh giá kết quả phân lớp, học viên sử dụng Phương pháp True/False Positive/Negative đã trình bày ở mục 2.5.2 của luận văn này để đánh giá nhầm đong đếm mô hình dự đoán tốt như thế nào trên từng lớp. Cụ thể trong bài toán phát hiện bất thường đậm độ trên hình ảnh X-Quang phổi của luận có 2 lớp (Abnormal và Normal) và việc phát hiện bất thường (Abnormal) là lớp dữ liệu nghiêm trọng hơn. Trong phương pháp này, ta thường định nghĩa lớp dữ liệu quan trọng hơn cần được xác định đúng là lớp **Positive**, lớp còn lại được gọi là **Negative**. Ta định nghĩa True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN) của bài toán luận văn đặt ra dựa trên confusion matrix đã chuẩn hóa như bảng Bảng 2-4:

**Xem xét kết quả kiểm tra mô hình trên 50 bức ảnh thuộc tập dữ liệu AB-Chest X-rays (chỉ xem xét kết quả của mô hình với quy tắc kết hợp FR3)**

Confusion matrix chưa chuẩn hoas dựa trên việc thống kê kết quả trên 50 bức ảnh thuộc tập dữ liệu AB-Chest X-rays được mô tả như bảng 4-9.

**Bảng 4-9:** Biểu diễn các giá trị TP, FP, TN, FN trên tập dữ liệu AB-Chest X-rays

	Dự đoán: Abnormal	Dự đoán: Normal
Thực tế: Abnormal	TP = 24	FN = 1
Thực tế: Normal	FP = 1	TN = 24

Ta tính toán các giá trị **TPR**, **FNR**, **FPR**, **TNR**, với **R-Rate** dựa trên *normalized confusion matrix* như bảng 4-10.

**Bảng 4-10:** Biểu diễn các giá trị TPR, FNR, FPR, TNR trên tập dữ liệu AB-Chest X-rays

	Dự đoán: Abnormal	Dự đoán: Normal
Thực tế: Abnormal	$TPR = 24/(24 + 1) = 0.96$	$FNR = 1/(24+1) = 0.04$
Thực tế: Normal	$FPR = 1/(1+24) = 0.04$	$TNR = 24/(1+24) = 0.96$

Độ chính xác của mô hình theo từng trường hợp cụ thể được tính như sau:

+ Độ chính xác tổng quát:

$$Accuracy = \frac{24 + 24}{24 + 1 + 24 + 1} = 0.96$$

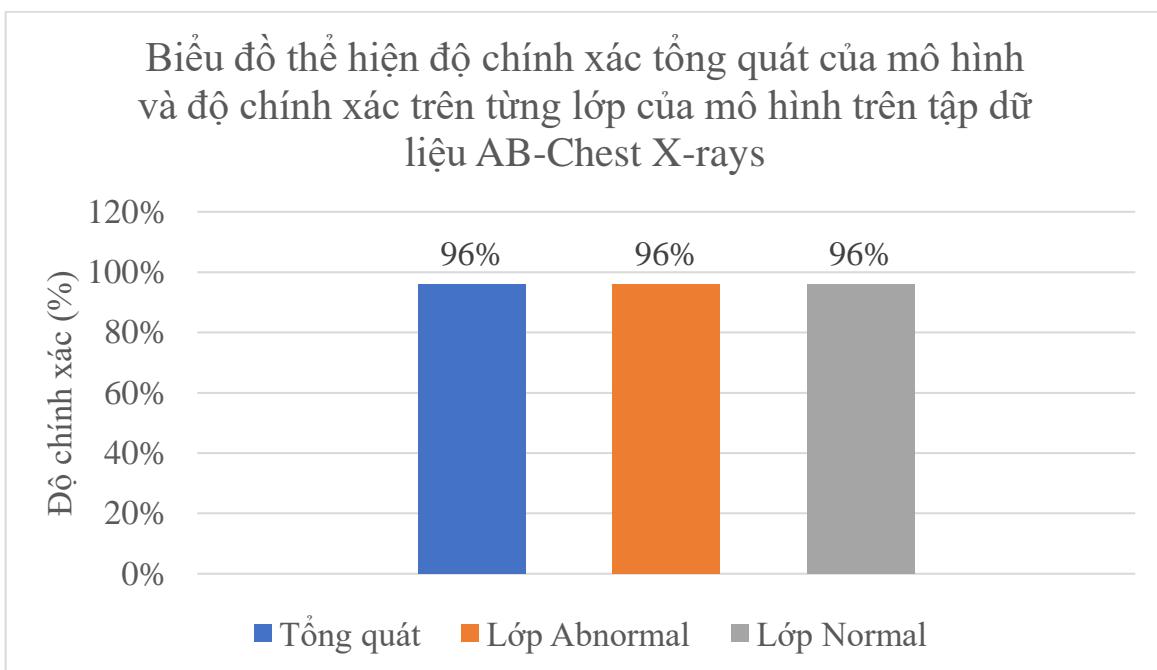
+ Khả năng phân loại Abnormal chính xác của mô hình (PPV):

$$PPV = \frac{24}{24 + 1} = 0.96$$

+ Khả năng phân loại Normal chính xác của mô hình (NPV)

$$NPV = \frac{24}{24 + 1} = 0.96$$

Với kết quả  $TPR = 0.96$  ta nhận thấy độ nhạy của mô hình đề xuất trên tập dữ liệu AB-Chest X-rays khá cao và tỷ lệ báo động nhầm thấp chỉ bằng 0.04. Với giá trị  $TNR = 0.96$  ta cũng có thể thấy được độ đặc hiệu của mô hình đề xuất trên tập dữ liệu này khá cao và tỷ lệ bỏ sót cũng thấp chỉ bằng 0.04.



**Hình 4-7:** Biểu đồ thể hiện độ chính xác tổng quát của mô hình và độ chính xác trên từng lớp của mô hình trên tập dữ liệu AB-Chest X-rays khi thống kê kết quả phân lớp của mô hình với quy tắc kết hợp FR3

**Xem xét kết quả kiểm tra mô hình trên 50 bức ảnh thuộc tập dữ liệu SZ-Chest X-rays (chỉ xem xét kết quả của mô hình với quy tắc kết hợp FR3)**

Confusion matrix chưa chuẩn hóa dựa trên việc thống kê kết quả trên 50 bức ảnh thuộc tập dữ liệu AB-Chest Xrays được mô tả như sau bảng 4-11.

**Bảng 4-11:** Biểu diễn các cái trị TP, FP, TN, FN trên tập dữ liệu SZ-Chest X-rays

	Dự đoán: Abnormal	Dự đoán: Normal
Thực tế: Abnormal	TP = 67	FN = 8
Thực tế: Normal	FP = 4	TN = 71

Ta tính toán các giá trị **TPR**, **FNR**, **FPR**, **TNR**, với **R-Rate** dựa trên **normalized confusion matrix** như bảng 4-12.

**Bảng 4-12:** Biểu diễn các cái trị TPR, FNR, FPR, TNR trên tập dữ liệu SZ-Chest X-rays

	Dự đoán: Abnormal	Dự đoán: Normal
Thực tế: Abnormal	$TPR = 67/(67 + 8) = 0.89$	$FNR = 8/(67+8) = 0.11$
Thực tế: Normal	$FPR = 4/(4+71) = 0.05$	$TNR = 71/(4+71) = 0.95$

Độ chính xác của mô hình theo từng trường hợp cụ thể được tính như sau:

+ Độ chính xác tổng quát:

$$Accuracy = \frac{67 + 71}{67 + 8 + 71 + 4} = 0.92$$

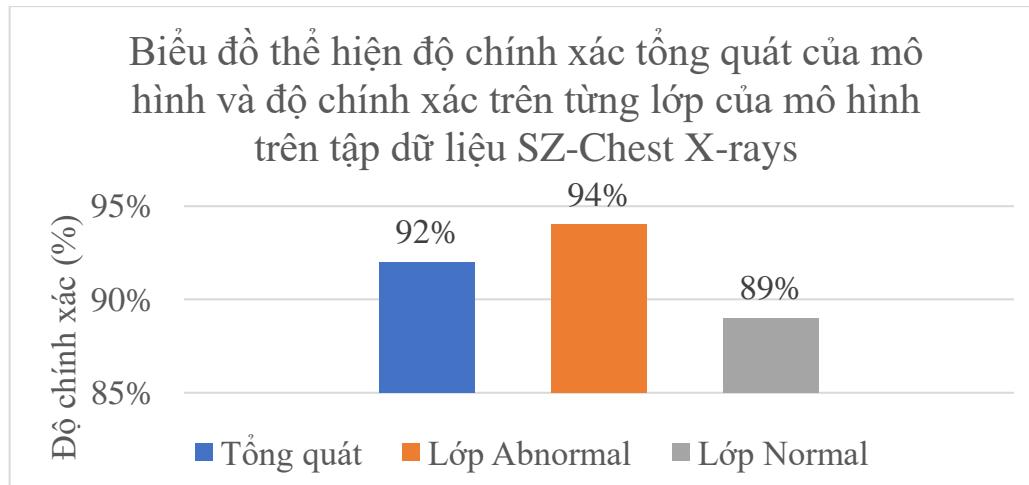
+ Khả năng phân loại lớp Abnormal chính xác của mô hình:

$$PPV = \frac{67}{67 + 4} = 0.94$$

+ Khả năng phân loại lớp Normal chính xác của mô hình:

$$NPV = \frac{71}{71 + 8} = 0.89$$

Với kết quả  $TPR = 0.89$  ta nhận thấy độ nhạy của mô hình để xuất trên tập dữ liệu SZ-Chest X-rays khá cao và tỷ lệ báo động nhầm chỉ bằng 0.11. Với  $TNR = 0.95$  ta cũng có thể thấy được độ đặc hiệu của mô hình để xuất trên tập dữ liệu này cũng khá cao và tỷ lệ bỏ sót cũng thấp chỉ bằng 0.05.



**Hình 4-8:** Biểu đồ thể hiện độ chính xác tổng quát của mô hình và độ chính xác trên từng lớp của mô hình trên tập dữ liệu SZ-Chest X-rays khi thống kê kết quả phân lớp của mô hình với quy tắc kết hợp FR3

Đối với tập dữ liệu SZ-Chest X-rays chưa được phân bổ để học trước đó, tuy nhiên độ chính xác tổng quát của mô hình (thống kê dựa trên quy tắc kết hợp FR3) là 92% và độ chính xác trên từng lớp Abnormal và Normal lần lượt là 94% và 89% cho thấy tính khả quan của mô hình đã đề xuất trong luận văn này và độ chính xác trên lớp Abnormal là 94% thể hiện mô hình đã đề xuất phù hợp với mục tiêu của luận văn là phát hiện bất thường đậm độ trên hình ảnh X-Quang.

## **CHƯƠNG 5 - KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### **5.1. Kết luận**

- Luận văn đã nghiên cứu về mạng học sâu phân lớp ảnh áp dụng cho bài toán phân lớp ảnh, các phương pháp và các tiêu chuẩn đánh giá hệ thống phân lớp ảnh về độ chính xác, thời gian huấn luyện, thời gian trung bình để phân lớp 1 bức ảnh.
- Luận văn tìm hiểu và mô hình hóa bài toán phát hiện bất thường đậm độ trên hình ảnh y khoa sử dụng kiến trúc mạng CNN, đây là một bài toán phân lớp hình ảnh nhị phân với ảnh đầu vào là hình ảnh x-quang phổi.
- Luận văn đã nghiên cứu và đề xuất mô hình Multi-CNNs và các quy tắc kết hợp dựa trên tập các trường hợp xảy ra của kết quả phân lớp nhằm tăng độ chính xác việc phân lớp. Từng CNN thành phần của mô hình được huấn luyện trên một bộ ảnh do học viên tự thu thập tại bệnh viện An Bình, TP. Hồ Chí Minh. Ngoài việc sử dụng các hình ảnh trong bộ dữ liệu đã thu thập để kiểm tra độ chính xác của mô hình, học viên cũng đã sử dụng hình ảnh của bộ dữ liệu quốc tế đã công bố để kiểm tra mô hình mạng. Kết quả đạt được của mô hình đề xuất là 96% đối với việc kiểm tra trên hình ảnh của bộ dữ liệu thu thập từ bệnh viện An Bình và 92% đối với bộ dữ liệu quốc tế. Qua đó cho thấy tính khả thi của mô hình đã đề xuất.

### **5.2. Hướng phát triển**

Học viên sẽ nghiên cứu để có thể song song hóa việc phân lớp của các thành phần trong mô hình đã đề xuất nhằm giảm thời gian tính toán của phân lớp của mô hình. Tiến hành thử nghiệm trên nhiều bộ dữ liệu khác để hiệu chỉnh và tăng độ chính xác của mô hình đề xuất.

Học viên sẽ tiếp tục huấn luyện mô hình với nhiều bộ dữ liệu khác nhau dựa trên kết quả đã đạt được của quá trình huấn luyện các CNN thành phần để nâng cao độ chính xác của mô hình nhằm hoàn thiện ứng dụng để có thể đưa vào kiểm nghiệm trên môi trường thực tế.

Phát triển bài toán cho việc phân lớp với multi-label (nhiều nhãn). Nghĩa là nếu là bất thường thì bất thường đó là biểu hiện của bệnh gì ở phổi.

Cài đặt mô hình trên một số nền tảng khác và thí nghiệm mô hình trên những hệ thống tính toán mạnh để có thể đánh giá được nhiều khía cạnh và từ đó tối ưu hóa mô hình.

## **DANH MỤC CÔNG TRÌNH CỦA TÁC GIẢ**

1. Phat Nguyen, Hai Tran, Thai Le, Tuan Le, Thuy Nguyen. "**Applying Multi-CNNs model for detecting abnormal problem on chest x-ray images.**" *Knowledge and Systems Engineering (KSE), 2018 10th International Conference on*. IEEE, 2018.

## **TÀI LIỆU THAM KHẢO**

### **1. Tiếng Việt**

- [1] Le Thi Thu, Hang (2016). Nghiên cứu về mạng neural tích chập và ứng dụng cho bài toán nhận dạng biển số xe. Luận văn Thạc sỹ, Trường Đại học công nghệ - Đại học quốc gia Hà Nội.
- [2] Nguyen Thi, Thu (2016), "Một số kỹ thuật phát hiện đối tượng bất thường trong ảnh y học". Luận văn Thạc sỹ, Trường Đại học Công nghệ Thông tin và Truyền thông.
- [3] Vu Manh, Hung (2016), "Nghiên cứu mô hình học sâu (deep-learning) và ứng dụng trong nhận dạng chữ viết tay". Luận văn Thạc sỹ, Trường đại học Thông tin và Truyền thông – Đại học Thái Nguyên.

### **2. Tiếng Anh**

- [1] Bottou, Léon. (2010), "Large-scale machine learning with stochastic gradient descent." Proceedings of COMPSTAT'2010. Physica-Verlag HD, pp. 177-186.
- [2] Daniel Lévy., Arzav Jain. (2016) "Breast Mass Classification from Mammograms using Deep Convolutional Neural Networks". arXiv preprint arXiv:1612.00542.
- [3] Dean, J., Corrado, G. S., Monga, R., Chen, K., Devin, M., Le, Q. V, ... Ng, A. Y. (2012). Large Scale Distributed Deep Networks. NIPS 2012: Neural Information Processing Systems, 1–11
- [4] Deng, Lih-Yuan. (2006), "The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning." pp. 147-148.
- [5] D.E. Rumelhart; G.E. Hinton and R.J. Williams (1986), Learning internal representations by error propagation, Rumelhart, D.E. et al. (eds.): Parallel distributed processing: Explorations in the microstructure of cognition (Cambridge MA.: MIT Press), 318-362.

- [6] Fakoor Rasool, et al, (2013) "Using deep learning to enhance cancer diagnosis and classification". Proceedings of the International Conference on Machine Learning.
- [7] Gallego-Posada, J. D, D. A. Montoya-Zapata, and O. L. Quintero-Montoya. Detection and Diagnosis of Breast Tumors using Deep Convolutional Neural Networks.
- [8] Gruetzemacher, Richard, and Ashish Gupta. "Using deep learning for pulmonary nodule detection & diagnosis." (2016).
- [9] He Yang, Hengyong Yu, Ge Wang (2016). "Deep Learning for the Classification of Lung Nodules". arXiv preprint arXiv:1611.06651.
- [10] J. Duchi, E. Hazan and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," Journal of Machine Learning Research, vol. 12, pp. 2121- 2159, 2011.
- [11] Puranikmath. Veena, Y. S. Lalita, and Shivaganga Patil (2015). "Automated Lung Nodule Detection and Classification using Morphological Segmentation and Texture". International Journal of Ethics in Engineering & Management Education, ISSN: 2348-4748, Volume 2, Issue 5, May 2015.
- [12] Soliman Omar S., and Eman AboElHamd (2014). Classification of Breast Cancer using Differential Evolution and Least Square Support Vector Machines. International Journal of Emerging Trends and Technology in Computer Science 3.2.
- [13] Tataru, Christine, et al. "Deep Learning for abnormality detection in Chest X-Ray images." (2017).
- [14] Zeiler, Matthew D. "ADADELTA: an adaptive learning rate method." arXiv preprint arXiv:1212.5701 (2012).
- [15] <https://vi.wikipedia.org/>
- [16] <https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10> - Ngày truy cập: 27/10/2018

- [17] <http://ruder.io/optimizing-gradient-descent/index.html#adagrad> - Ngày truy cập: 21/11/2018
- [18] Japanese Society of Radiological Technology: <https://www.jsrt.or.jp/>
- [19] National Institutes of Health U.S. National Library of Medicine: <https://ceb.nlm.nih.gov/repositories/tuberculosis-chest-x-ray-image-data-sets/> - ngày truy cập: 13/10/2018
- [20] ConvNetJS is a Javascript library - <https://cs.stanford.edu/people/karpathy/convnetjs/>
- [21] <https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html> - ngày truy cập: 13/10/2018

# PHỤ LỤC

## 1. Mình chúng công trình đã được công bố

Friday, November 2 12:20 - 13:30

Lunch break 

Friday, November 2 13:30 - 15:20

Special Session #1: Intelligent Software and Knowledge Representation 

Room: BALLROOM 2

Chairs: Nhon Do (Computer Science, Vietnam), Hien Nguyen (University of Information Technology, Vietnam)

Nowadays there are many practical applications in Artificial Intelligence and the Internet. The Intelligent Software has been created to better serve the increasing needs of the people. These kinds of software have been applied in a wide range of areas such as Expert Systems, Intelligent Problems Solver, Decision Support Systems and another Intelligent Systems. Knowledge Representation (KR) and Reasoning are exciting and well-established fields of research in Intelligent Software. KR is key drivers of innovation in computer science. In recent years, KR has also derived challenges from new and emerging fields including the semantic web, computational biology, and the development of software agents.

This special session aims to bring together researchers and developers with all these different backgrounds to discuss and exchange new ideas and the latest achievements in building Intelligent Software and methods for KR

### A Novel Method for Time Series Anomaly Detection based on Segmentation and Clustering

[Huynh Thi Thu Thuy](#) (Ton Duc Thang University, Vietnam); [Tuan Anh Duong](#) (Ho Chi Minh City University of Technology, Vietnam); [Vo Thi Ngoc Chau](#) (HCMUT, Vietnam)

### Assessment of georeferencing methods on MODIS Terra/Aqua and VIIRS NPP satellite images in Vietnam

[Ha Pham Van](#) (VNU University of Engineering and Technology, Vietnam)

### Criteria of a Knowledge model for an Intelligent Problems Solver in Education

[Hien Nguyen](#) (University of Information Technology, Vietnam); [Nhon Do](#) (Computer Science, Vietnam); [Nha Tran](#) (University of Transport and Communications, Campus in Ho Chi Minh City, Vietnam); [Hau Pham](#) (Quang Binh University, Vietnam)

### Combining Fuzzy Set - Simple Additive Weight and Comparing With Grey Relational Analysis For Student's Competency Assessment In The Industrial

[Lich Nghiem](#) (Thuongmai University, Vietnam); [Bui Cuong](#) (Institute of Mathematics, Hanoi, Vietnam); [Ha Dinh](#) (Thuongmai, Vietnam)

### Applying Multi-CNNs model for detecting abnormal problem on chest x-ray images

[Phat Nguyen](#) (Nguyen Tat Thanh University, Vietnam); [Hai Tran](#) (HCM University of Pedagogy, Vietnam); [Le Hoang Thai](#) (University of Science, Ho Chi Minh city., Vietnam); [Tuan Le](#) (Nguyen Tat Thanh University, Vietnam); [Nguyen Thanh Thuy](#) (VNU Ha Noi, Vietnam)

Friday, November 2 13:30 - 15:35

Special Session #3: Deep Learning and Applications to Language Processing 

Room: BALLROOM 3

Chairs: Ngo Xuan Bach (Posts and Telecommunications Institute of Technology, Vietnam), Nguyen Minh Tien (Hung Yen University of Technology and Education, Vietnam)

The session focuses on developing deep learning models and applications to real-world problems relevant to the field of language processing. Deep learning has made many breakthroughs in various areas of computer science such as computer vision, speech, and natural language processing (NLP). In the last few years, various models have been proposed for different data representations make deep learning transited successfully from research to applications. In the context of this session, we will explore and discuss both new models and applications of deep learning in natural, programming language, and speech processing.

### Variants of Long Short-Term Memory for Sentiment Analysis on Vietnamese Students' Feedback Corpus

[Vu Duc Nguyen](#), [Kiet Van Nguyen](#) and [Ngan L.T. Nguyen](#) (University of Information Technology, VNU-HCM, Vietnam)

### A study on integrating distinct classifiers with bidirectional LSTM for Slot Filling task

[Phong Do](#) (Vietnam National University, Hanoi, Vietnam); [Minh Le Nguyen](#) (Japan Advanced Institute of Science and Technology, Japan)

### Combining Advanced Methods in Japanese-Vietnamese Neural Machine Translation

2018 10th International Conference on Knowledge and Systems Engineering (KSE)

# Applying Multi-CNNs model for detecting abnormal problem on chest x-ray images

Phat Nguyen Kieu  
Department of Academic Affairs,  
Nguyen Tat Thanh University  
Ho Chi Minh City, Viet Nam  
[nkphat@ntt.edu.vn](mailto:nkphat@ntt.edu.vn)

Tuan Le  
Informatics Center,  
Nguyen Tat Thanh University  
Ho Chi Minh City, Viet Nam  
[lhvtuan@ntt.edu.vn](mailto:lhvtuan@ntt.edu.vn)

Hai Son Tran  
HCM University of Education,  
University of Informatics Technology- VNU-HCM,  
Ho Chi Minh City, Vietnam  
[laits@hcme.edu.vn](mailto:laits@hcme.edu.vn)

Thuy Thanh Nguyen,  
University of Technology - VNUH  
Ha Noi city, Vietnam  
[nguyenthanhthuy@vnu.edu.vn](mailto:nguyenthanhthuy@vnu.edu.vn)

Thai Hoang Le  
Department of Computer Science,  
University of Science - VNUHCM  
Ho Chi Minh City, Vietnam  
[jihai@fit.hcmus.edu.vn](mailto:jihai@fit.hcmus.edu.vn)

**Abstract—** Image diagnosis is the significant problem in medicine. Nowadays, with modern facilities that allow doctors to diagnose early and accurately disease, limiting unnecessary treatment procedures. By that way, the image diagnosis is at the forefront of the processing diagnosis and treatment of the disease. Heart and lung failure accounts for more than 500,000 deaths annually in the United States and is most commonly screened for using plain film chest X-Ray (CXR). With the growing number of patients, the doctors must overwork, so he cannot counsel and direct take care of his patient. So, a computer system that supports image classification is needed. In this paper, we propose a deep learning model to detect abnormal sentis in chest x-ray images. The proposed model uses multiple Convolutional Neural Network to decide input image, this is called Multi-CNNs. Input data is the digital chest X-ray image dataset that was collected from 6/2017 to 3/2018 at An Binh Hospital, HCM, VN (AB-CXR-Database). Each component of the Multi-CNN is a convolutional neural network that is developed base on ConvnetJS library. The output of the proposed model is Normal/Abnormal density. In this paper, we also propose a method for synthesizing the results of the components of the model which we are called Fusion rules. The experimental results 96% in our x-rays image dataset showed the feasibility of a proposed Multi-CNNs model.

**Keywords—** image classification, Chest x-ray image, Convolutional neural network (CNN), chest x-ray image classification using CNN

## I. INTRODUCTION

Heart and lung failure accounts for more than 500,000 deaths annually in the United States and is most commonly screened for using plain film chest X-Ray (CXR)[2]. In Vietnam and most of the countries in the world, x-ray and CT images are used much in the cancer sign diagnosis. Chest x-ray images, or radiographs, provide a single view of the chest cavity. CT scans can provide a complete view of the chest internals and can thus be used to more easily detect shape, size, location, and density of lung nodules [3]. However, CT scan technology is expensive and is often not available in smaller hospitals or rural areas. By contrast, basic chest radiographs are relatively cheap and fast and expose the patient too little radiation, so they are usually the first diagnostic step for detecting any chest abnormalities [3].

In recent years, machine learning has been used in the detection and classification of medical images to assist in

the early detection of expression pathology. Especially, it's can help the radiologist reduce workload. There are many classification methods is proposed to solve this problem. At present, popular methods for solving image classification problems, such as K-Mean, K-NN, deep neural network, Support Vector Machine (SVM)... One of the popular approaches is used method of Artificial Neural Networks for pattern classification problem. Convolutional neural network (CNN)[1] is one of the deep learning models that has garnered much interest from researchers in recent years. It's used a lot of in image classification, image recognition, language translate, medical diagnostics, and many other domains, etc. and giving a result with high accuracy.

Therefore, in this paper, we propose a model called Multi-CNNs based on convolutional neural network. In this paper, we also propose a method for synthesizing the results of the components of the model which we are called Fusion rules. The experimental results 96% in our x-rays image dataset showed the feasibility of a proposed Multi-CNNs model.

## II. BACKGROUND AND RELATED WORK

### A. Background

Deep Learning is a new area of Machine Learning research. Deep learning architectures such as deep neural networks, deep belief networks and recurrent neural networks have been applied to fields including computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, and bioinformatics, etc [4], which give a result with high accuracy.

Convolutional neural networks (CNN) is a special architecture of artificial neural networks, proposed by Yann LeCun in 1988 that has been widely applied to a variety of pattern recognition problems, such as computer vision, speech recognition, image classification, image recognition, language translate, medical diagnostics, etc. A CNN is composed of layers that filters (convolve) the inputs to get useful information. These convolutional layers have parameters (kernel) that are learned so that these filters are adjusted automatically to extract the most useful information for the task at hand without feature selection. CNN is better to work with images. Normal Neural networks do not fit well with image classification problems.

ConvNetJS is an open JavaScript library for training Deep Learning models (Deep Neural Networks) entirely in your browser[5]. We used it to build an application for training and testing Multi-CNNs model which we proposed in this paper.

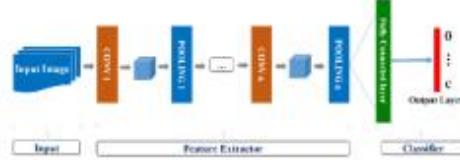


Figure 1: CNN Image Classifier Model

Figure 1: shows an overview of convolutional neural network architecture for image classification

#### B. Related work

The problem of detecting abnormal objects in medical imaging is a great practical problem to support physicians in diagnosing diseases in medicine, the more accurate the detection results, the more effective in the process of diagnosis and treatment of doctors. Using deep learning methods for detecting abnormal problem on medical images that has attracted many interests from researchers. In [2], developing a simple preprocessing pipeline using digital image processing techniques and expert radiologist advice; Create a pipeline that can apply three neural network architectures that have proven successful in classification tasks, GoogLeNet, InceptionNet, and ResNet, on our cohort of CXR images; Use neural network visualization techniques to understand what type of features our model weights most heavily. In [3], this study uses a revolutionary image recognition method, deep learning, for the classification of potentially malignant pulmonary nodules. They report results of their initial findings and compare performance of deep neural nets using a combination of different network topologies and optimization parameters. Classification accuracy, sensitivity and specificity of the network performance are assessed for each of the four topologies. In [6], they propose a novel work to first segment CT scan image to obtain all probable nodule candidates. Then they perform SVM based classification of the candidates based on image moments and texture features. Potential nodule candidates are marked, and other candidates are omitted. They perform the test on 50 images from Tata Cancer Research Centre CT images. Also compared the results of proposed technique with nearest neighbor classifier. It was evident that by taking statistical features from GLCM matrix, performance of the nearest neighbor classifier improved significantly. However SVM performance was observed to be consistent. Out of 50 images with nodules, their system detected all fifty nodules correctly whereas miss classifying 8 candidates in total. False positive was about 6% whereas false negative was zero. Involves matching these features to yield a result that is visually similar.

### III. MULTI-CNNs MODEL FOR DETECTING ABNORMAL SENTISY IN CHEST X-RAY IMAGE

#### A. Multi-CNNs model architecture

As mentioned above, in this paper we propose a model that is called Multi-CNNs. This model consists of three components, each of which is a CNN (Figure 2). For each component, we call them the following: CNN-128F, CNN-64L, CNN-64R.

All CNN components were run on a PC with specification: Intel Core i7 - 7700, 8 GB RAM, Windows 10 pro 64bit, Google Chrome browser.

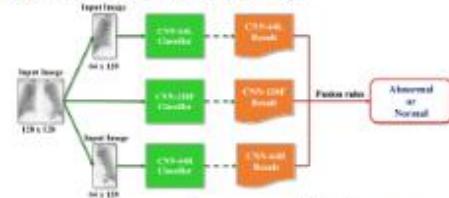


Figure 2: Multi-CNNs Model architecture

With each input image, the result of each CNN classifier of the model component is normal or abnormal and a probability value and then we use a rule to compute those results which we called Fusion Rule. It mentioned in the section III.C.

#### B. CNN components of Multi-CNNs Model

##### a) CNN-128F

CNN-128F, which has an architecture as Figure 3 used to train on 128 x 128 chest x-ray image dataset (Dataset 1). This dataset will mention in section IV.A.

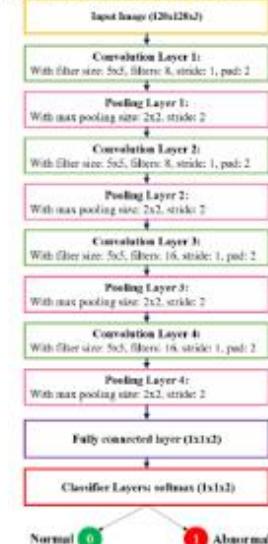


Figure 3: CNN-128F architecture

Figure 3: shows an overview architecture of the CNN-128F component which is part of the Multi-CNNs Model.



Figure 4: Testing accuracy and Training accuracy

#### b) CNN-64L and CNN-64R

CNN-64L and CNN-64R, which has an architecture as Figure 5 used to train on 64 x 128 left and right chest x-ray image dataset (Dataset 2, Dataset 3). Those datasets will mention in section IV.A.

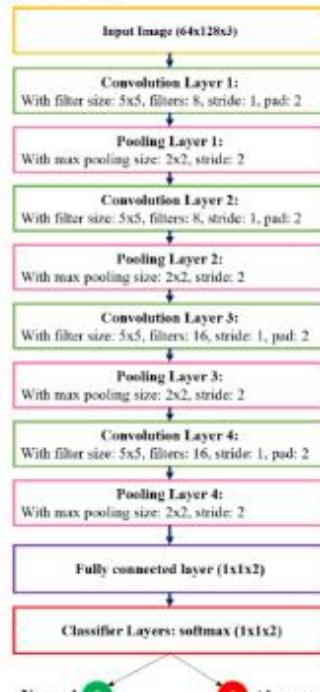


Figure 5: CNN-64L, CNN-64R architecture

Figure 5: shows an overview architecture of the CNN-64L and CNN-64R component which is part of the Multi-CNNs Model.

CNN-128F, CNN-64L, CNN-64R performance was assessed using a held-out test set with 100 independent images consist of 50 normal and 50 abnormal chest X-Ray

images. Overfitting was assessed by comparing the training accuracy and testing accuracy that is showed as Figure 4.

#### c) CNN components accuracy evaluation

Assess the accuracy of the classification is very important, because it allows to predict the accuracy of the classification results of future data. Accuracy helps to compare different classifiers. With each CNN classifiers component of Multi-CNNs Model, we used Hold-out method for evaluating.

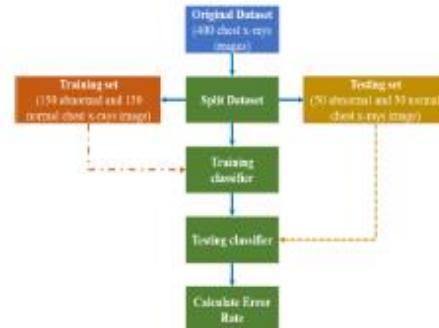


Figure 6: The Hold-out method for evaluating CNN component

Figure 6: shows an overview of the Hold-out method which we had used for evaluating the accuracy of CNN-128F, CNN-64L, CNN-64R components of Multi-CNNs Model

Table 1: Training and testing result for each CNN classifier component:

Network Result	CNN-128F	CNN-64L	CNN-64R
Training time (minutes)	360	25	150
Max Accuracy on test set (%)	94	93	93
Classification loss (%)	1	10.5	4
L2 Weight loss (%)	1	1.8	2

#### C. Fusion rules base on Conclusion tree

As mentioned at section III.A, in Multi-CNNs model, we have used an association rule to compute these results which we called **Fusion Rule**. In this section, we describe detail about it.

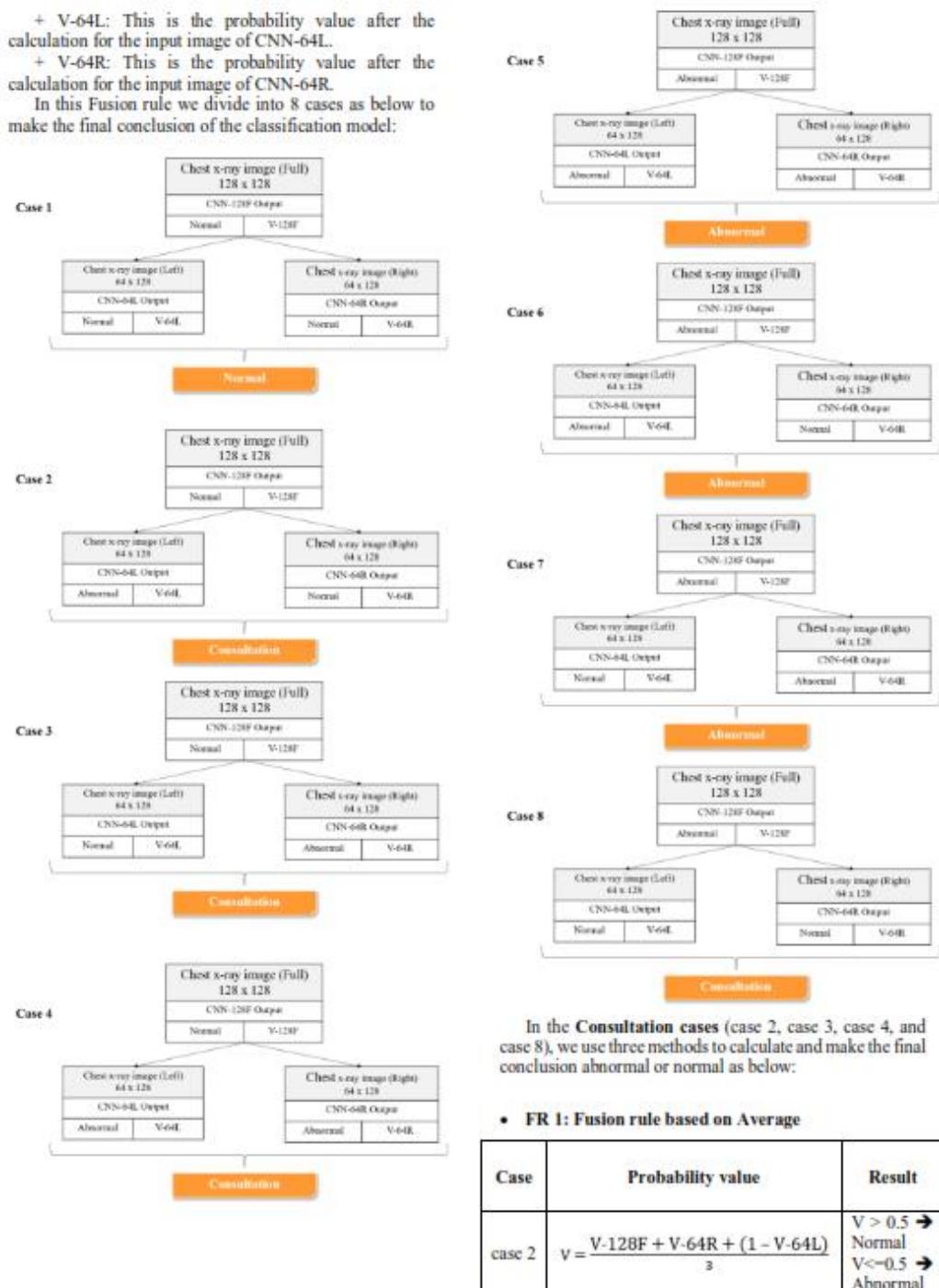
For each CNN output, we derive the result consist of two values, one of which is the input image of the class, and the probability value of that image belongs to. We call the probability value as follows:

+ V-128F: This is the probability value after the calculation for the input image of CNN-128F.

+ V-64L: This is the probability value after the calculation for the input image of CNN-64L.

+ V-64R: This is the probability value after the calculation for the input image of CNN-64R.

In this Fusion rule we divide into 8 cases as below to make the final conclusion of the classification model:



In the **Consultation cases** (case 2, case 3, case 4, and case 8), we use three methods to calculate and make the final conclusion abnormal or normal as below:

- **FR 1: Fusion rule based on Average**

Case	Probability value	Result
case 2	$V = \frac{V-128F + V-64R + (1 - V-64L)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal

case 3	$V = \frac{V-128F + V-64L + (1-V-64R)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 4	$V = \frac{V-128 + (1-V-64R) + (1-V-64L)}{3}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 8		$V \leq 0.5 \rightarrow$ Normal $V > 0.5 \rightarrow$ Abnormal

- FR 2: Fusion rule based on overall

Case	Probability value	Result
case 2	$V = \frac{V-128F + V-64R + (1-V-64L)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 3	$V = \frac{V-128F + V-64L + (1-V-64R)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 4	$V = \frac{V-128 + (1-V-64R) + (1-V-64L)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 8		$V \leq 0.5 \rightarrow$ Normal $V > 0.5 \rightarrow$ Abnormal

- FR 3: Fusion rule based on conflict regions

Case	Probability value	Result
case 2	$V = \frac{V-128F + (1-V-64L)}{2}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 3	$V = \frac{V-128F + (1-V-64R)}{2}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 4	$V = \frac{2*V-128 + (1-V-64L) + (1-V-64R)}{4}$	$V > 0.5 \rightarrow$ Normal $V \leq 0.5 \rightarrow$ Abnormal
case 8		$V \leq 0.5 \rightarrow$ Normal $V > 0.5 \rightarrow$ Abnormal

For example:

Tested with image named abnormal\_test\_4\_full.jpg:

$$\begin{aligned} &+ \text{CNN-128F: Normal} & V-128F = 0.98609 \\ &+ \text{CNN-64L: Abnormal} & V-64L = 0.92142 \end{aligned}$$

$$+ \text{CNN-64R: Abnormal} \quad V-64R = 0.99969$$

$$\text{FR1: } V = 0.354993333 \rightarrow \text{Abnormal}$$

$$\text{FR2: } V = 0.266245 \rightarrow \text{Abnormal}$$

$$\text{FR3: } V = 0.5127675 \rightarrow \text{Normal}$$

#### IV. EXPERIMENTS AND DISCUSSION

##### A. Dataset

For this project, our primary dataset is chest x-ray image of patients that was collected from 6/2017 to 3/2018 at An Binh Hospital, HCM, VN (AB-CXR-Database). The dataset contains 400 digital images were labelled, which we divide into a training set with 300 images, a testing set with 100 images. For each image, it was labelled 0 for normal, 1 for abnormal by an expert who has 15 years experience in the field of imaging diagnosis. We have build 03 datasets from images which collected:

- **Dataset 1:** Containing 400 chest x-ray images with dimension 128x128.



- **Dataset 2:** Containing 400 left chest x-ray images with dimension 64x128.



- **Dataset 3:** Containing 400 right chest x-ray images with dimension 64x128.



##### B. Methods

- Pre-processing image

With each original chest x-ray image has been labeled, we scaled it to 128 x 128 size. And then we divided it into two parts to get 3 input images including left chest x-ray, right chest x-ray and full chest x-ray image. From this, we have constructed three datasets as mentioned above.

- Training Multi-CNNs Model's

To train the model, we trained each CNN component of the model with the CNN architecture mentioned in Section III with the following parameters:

- Learning Rate: 0.01, Momentum: 0.9,
- Weight decay: 0.0001, Batch size: 1,
- Method for training is Adadelta.

- Multi-CNNs Model's accuracy evaluation

As mentioned above, assessing the accuracy of the classification model is very important. To evaluate the model that we have proposed, we have selected any 50 chest X-ray images (*consist of 25 labelled abnormal and 25 labelled normal images*) from [10, 11] for model testing which we proposed. The results of the 96% accuracy evaluation that we will present are detailed in the next section.

### C. Results

#### a) The result of Multi - CNNs Model testing:

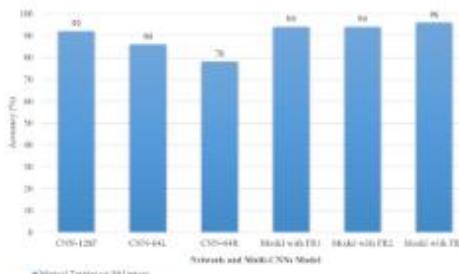


Figure 6: Result of testing on 50 chest X-Ray images

Table 2: Result of each CNN component of model on 50 chest x-ray images:

Network	CNN-128F		CNN-64L		CNN-64R	
	Amo unt	(%)	Amo unt	(%)	Amo unt	(%)
True	46	92	43	86	39	78
False	4	8	7	14	11	22

Table 3: Result of Multi-CNN model with 3 methods computes:

Fusion rule	Multi-CNNs Model results					
	G1		G2		G3	
	Amo unt	(%)	Amo unt	(%)	Amo unt	(%)
True	47	94	47	94	48	96
False	3	6	3	6	2	4

### V. CONCLUSION

In this research, we proposed Multi-CNNs model and a method for synthesizing the results of the components of the model which we are called fusion rules.

The proposed Multi-CNNs model consists of three components: CNN 128F, CNN 64L, and CNN 64R. These components are developed based on CNN.

The proposed model used the association fusion rule in order to combine the results. The fusion rules process to integrate the results in 8 cases to make the final conclusion of the Multi-CNNs classification model. In 8 cases of this

rule, we tested three combined methods for cases with unclear conclusions.

We tested Multi-CNNs model with Fusion rules to detect abnormal sentis on 50 chest X-ray images (*consist of 25 labelled abnormal and 25 labelled normal images*) which were chose from [10, 11]. The accuracy results 96% showed the feasibility of the proposed model when it combines this rule.

### VI. ACKNOWLEDGEMENT

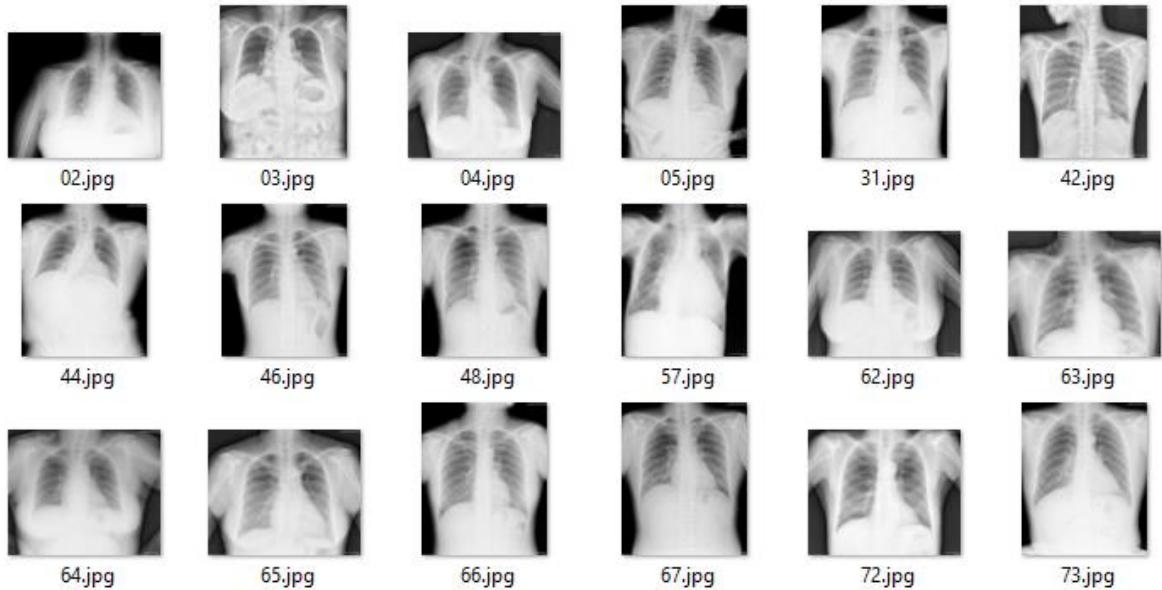
We would like to thank doctor Chung Gia Vien for helping us with labelling chest X-ray images. We also thank the Diagnostic Department of An Binh Hospital, HCM has allowed us to use database chest X-ray image database of the hospital to perform this study.

### VII. REFERENCES

- [1] A. Krizhevsky, I. Sutskever and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, pp. 1097-1105, 2012
- [2] Tataru, Christine, et al. "Deep Learning for abnormality detection in Chest X-Ray images." (2017).
- [3] Graetzemacher, Richard, and Ashish Gupta. "Using deep learning for pulmonary nodule detection & diagnosis." (2016).
- [4] Ghasemi, F., Mehrdadluvi, AR.; Fassih, A.; Perez-Sanchez, H. (2017). "Deep Neural Network in Biological Activity Prediction using Deep Belief Network".
- [5] A.Karpathy, "ConvNetJS: Deep learning in your browser," <http://cs.stanford.edu/people/karpathy/convnetjs>, 2014.
- [6] Puranikmath, Veena, Y. S. Lalita, and Shivangana Patil (2015). "Automated Lung Nodule Detection and Classification using Morphological Segmentation and Texture". International Journal of Ethics in Engineering & Management Education, ISSN: 2348-4748, Volume 2, Issue 5, May 2015.
- [7] Yang, He, Hengyong Yu, and Ge Wang. "Deep learning for the classification of lung nodules," arXiv preprint arXiv:1611.06651 (2016).
- [8] Bhandare, Ashwin, et al. "Applications of Convolutional Neural Networks." International Journal of Computer Science and Information Technologies (2016): 2206-2215.
- [9] Vo, An Tien, Hai Son Tran, and Thai Hoang Le. "Advertisement image classification using convolutional neural network." Knowledge and Systems Engineering (KSE), 2017 9th International Conference on. IEEE, 2017.
- [10] JSRT Database - <http://db.jsrt.or.jp/eng.php> - download date: 17.5.2018
- [11] Dataset from Indiana University hospital network - [https://openi.nlm.nih.gov/tmes/collections/NLMCXR\\_png.tgz](https://openi.nlm.nih.gov/tmes/collections/NLMCXR_png.tgz) - Download date: 17.5.2018
- [12] Nhon Do, Hien Nguyen, A reasoning method on Computational Network and Its applications, 2011 International Multi Conference of Engineers and Computer Scientists, IMECS 2011, ISBN: 978-988-18210-3-4, pp. 137-141, Hongkong , March 2011
- [13] Tran, Hai Son, Thai Hoang Le, and Thuy Thanh Nguyen. "The Degree of Skin Burns Images Recognition using Convolutional Neural Network." Indian Journal of Science and Technology9.45 (2016).
- [14] Amato, Filippo, et al. "Artificial neural networks in medical diagnosis." (2013): 47-58.

## 2. Bộ dữ liệu hình ảnh

### 2.1. Hình ảnh thu thập từ bệnh viện An Bình, TP. Hồ Chí Minh



### 2.2. Bộ hình ảnh quốc tế

