**BDRPC187TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

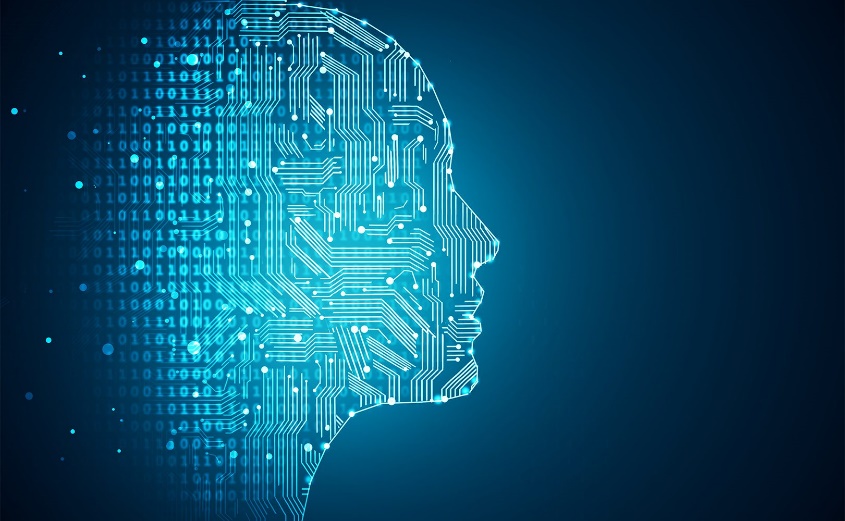
**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**BÁO CÁO**

**TIỂU LUẬN CHUYÊN NGÀNH**

*ĐỀ TÀI:*

**ỨNG DỤNG THUẬT TOÁN MÁY HỌC TRONG VIỆC XỬ LÝ ẢNH ĐỂ CHẨN ĐOÁN BỆNH PHỔI TRONG Y KHOA**

****

**Giảng viên: TS. Vũ Quang Huy**

* **Phạm Hồng Cang 15110014**
* **Lê Minh Chương 15110020**

**Nhận xét của giảng viên hướng dẫn**

**Chữ ký của giảng viên hướng dẫn**

MỤC LỤC

[I. Giới thiệu 5](#_Toc532333756)

[1. Machine Learning là gì? 5](#_Toc532333757)

[2. Lí do chọn đề tài 5](#_Toc532333758)

[3. Ý nghĩa đề tài 7](#_Toc532333759)

[II. Mục tiêu 8](#_Toc532333760)

[III. Quá trình thực hiện 8](#_Toc532333761)

[1. Đặt vấn đề 8](#_Toc532333762)

[2. Chuẩn bị dữ liệu 9](#_Toc532333763)

[3. Chọn thuật toán 9](#_Toc532333764)

[4. Training model 10](#_Toc532333765)

[5. Kiểm thử 11](#_Toc532333766)

[IV. Cơ sở lý thuyết 11](#_Toc532333767)

[1. Xử lý ảnh 11](#_Toc532333768)

[2. Neural Networks 11](#_Toc532333770)

[3. Convolutional Neural Networks – CNN 15](#_Toc532333771)

[V. Quá trình triển khai xử lý bài toán 25](#_Toc532333772)

[1. Flow chart 25](#_Toc532333773)

[2. Xử lý dữ liệu 25](#_Toc532333774)

[3. Huấn luyện 30](#_Toc532333775)

[VI. Kết quả và đánh giá: 31](#_Toc532333776)

[VII. Kế hoạch thực hiện 32](#_Toc532333777)

[Tài liệu tham khảo 32](#_Toc532333778)

MỤC LỤC HÌNH

[Hình 1: Phổi bình thường 9](#_Toc532333623)

[Hình 2: Phổi bị bệnh 9](#_Toc532333624)

[Hình 3: Kiến trúc mạng Neural Networks 12](#_Toc532333625)

[Hình 4: Quá trình xử lý thông tin 12](#_Toc532333626)

[Hình 5: Quá trình xử lý thông tin với hàm 13](#_Toc532333627)

[Hình 6: ANN (Artificial Neural Network) với cấu trúc Fully – connected 14](#_Toc532333628)

[Hình 7: Mô hình phép toán tích chập 15](#_Toc532333629)

[Hình 8: Hình ảnh sau khi lọc 17](#_Toc532333630)

[Hình 9: Ma trận đầu vào được bao quanh bởi đường viền phụ kích thước p 18](#_Toc532333631)

[Hình 10: Một ảnh sau khi được Convolution 19](#_Toc532333632)

[Hình 11: Ví dụ pooling theo giá trị cực đại 20](#_Toc532333633)

[Hình 12: So sánh sau khi Max Pooling từ ảnh đã được Convolution 20](#_Toc532333634)

[Hình 13: Hàm kích hoạt 21](#_Toc532333635)

[Hình 14: Một Neural Network đơn giản áp dụng ReLU 22](#_Toc532333636)

[Hình 15: Cách hoạt động của Fully-Connected 23](#_Toc532333637)

[Hình 16: Quá trình hoạt động của mạng CNN 23](#_Toc532333638)

[Hình 17: Quá trình triển khai xử lý bài toán 25](#_Toc532333639)

[Hình 18: Một ảnh phổi sau khi được Convolution 27](#_Toc532333640)

[Hình 19: So sánh sau khi Max Pooling từ ảnh đã được Convolution 28](#_Toc532333641)

[Hình 20: Sau khi training bộ dữ liệu 31](#_Toc532333642)

[Hình 21: Bảng kế hoạch thực hiện 32](#_Toc532333643)

# Giới thiệu

## Machine Learning là gì?

Machine Learning là một lĩnh vực nhỏ của Khoa Học Máy Tính, nó có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình cụ thể. Machine Learning gây nên cơn sốt công nghệ trên toàn thế giới trong vài năm nay. Trong giới học thuật, mỗi năm có hàng ngàn bài báo khoa học về đề tài này. Trong giới công nghiệp, từ các công ty lớn như Google, Facebook, Microsoft đến các công ty khởi nghiệp đều đầu tư vào Machine Learning.

Có khá nhiều ứng dụng của công nghệ học máy đã được áp dụng trong cuộc sống hàng ngày như: Google dịch, xe ô tô tự lái, hệ thống gợi ý mua hàng, hệ thống nhận diện khuôn mặt... Đặc biệt, Machine Learning là 1 xu hướng phát triển nhanh chóng trong ngành chăm sóc sức khỏe, nhờ vào sự ra đời của các thiết bị và máy cảm ứng đeo được sử dụng dữ liệu để đánh giá tình hình sức khỏe của bệnh nhân trong thời gian thực (real-time). Công nghệ Machine Learning còn giúp các chuyên gia y tế xác định những xu hướng hoặc tín hiệu để cải thiện khả năng điều trị, chẩn đoán bệnh.

## Lí do chọn đề tài

1. Hiện tại
   * Ngành y khoa
     + Nhu cầu hỗ trợ bác sĩ trong y khoa càng lúc càng tăng theo thời gian. Việc đọc, phân tích hình ảnh và liên hệ bệnh lý cần đòi hỏi bác sĩ có kiến thức rộng và kinh nghiệm nhiều. Đọc thủ công không những mất rất nhiều thời gian mà lại có độ tin cậy không cao.
     + Số lượng bệnh nhân càng lúc càng tăng và các loại bệnh cơ bản lặp đi lặp lại nhiều nên cần có công cụ hỗ trợ các bác sĩ chuẩn đoán nhanh các bệnh cơ bản đó.
     + Y tế được biết đến là dịch vụ về chăm sóc sức khỏe con người cũng như là một trong những dịch vụ yêu cầu bắt buộc phải phát triển trên mức kinh tế để có thể đảm bảo cuộc sống cho người dân cũng như là sự tồn vong của nhân loại. Những ứng dụng của khoa học công nghệ luôn chú trọng vào việc có thể ứng dụng vào y tế rất nhiều và được ưu tiên hàng đầu.
     + Thế kỷ XXI là một thế kỷ của sự hội nhập kinh tế. Vì thế mà Việt Nam cần đẩy mạnh về việc có thể giao lưu kinh tế với các quốc gia trong và ngoài khu vực. Để có thể đảm bảo về cơ sở vật chất, dịch vụ thì việc có được những chiến lược, những áp dụng công nghệ, khoa học vào y tế là điều cần thiết. Đây được xem là một trong những lĩnh vực có thể thu hút được đầu tư từ các đơn vị nước ngoài.
     + Trong kỷ nguyên mới ngành y tế càng cần thiết để con người có được sức khỏe tốt nhất cống hiến cho xã hội. Ngành y tế ứng dụng các thiết bị khoa học để phát triển hơn, khi sức khỏe con người trở nên ổn định thì đương nhiên mọi hoạt động kinh tế, xã hội đều được thúc đẩy mạnh mẽ.
   * Bệnh phổi
     + Vì quá trình công nghiệp hóa hiện đại hóa đang diễn ra với tốc độ nhanh chóng, từ đó kéo theo rất vấn đề như: ô nhiễm môi trường đặc biệt là không khí … Dẫn đến các bệnh liên quan về đường hô hấp gia tăng và ngày càng phổ biến. Nên có 1 lượng dữ liệu lớn về bệnh phổi - sau khi chuẩn đoán (bằng phương pháp chụp X - ray).
     + Theo số liệu thống kê, năm 2015 có 1,8 triệu người bị chết do lao phổi, trong số 10,4 triệu người mắc bệnh.
     + Theo dự đoán của WHO, đến năm 2020, bệnh phổi sẽ là nguyên nhân tử vong xếp hàng thứ 3 trên toàn thế giới và là nguyên nhân thứ 5 trong các bệnh gây ra tàn phế.
2. Tương lai
   * Machine Learning nói riêng và AI nói chung trong tương lai sẽ phát triển rất nhanh chóng và hỗ trợ, phục vụ con người trong nhiều lĩnh vực như: kinh tế, công nghiệp, đời sống, giáo dục… và đặc biệt là y khoa.
   * Machine Learning giúp phổ cập rộng rãi các hệ thống hồ sơ y tế điện tử, cải thiện tốc độ xử lý chẩn đoán.
   * Khi triệu chứng bệnh xuất hiện, có thể có hàng triệu nghiên cứu và kết quả liên quan đến chúng và có thể tốn rất nhiều thời gian để phân tích một lượng dữ liệu lớn như vậy. Machine Learning sẽ vượt trội hơn so với cách truy vấn dữ liệu truyền thống, để quét, phân tích và tóm tắt các văn bản y tế giúp các bác sĩ tập trung vào chẩn đoán bệnh và có thể đưa ra kết luận trong thời gian sớm nhất.
   * Mô hình sử dụng công nghệ sẽ nhanh hơn và chính xác hơn so với mô hình được hỗ trợ bởi máy tính truyền thống và có thể tự cải thiện khả năng phát hiện và phân loại bệnh của chúng thông qua khả năng tự học hỏi và nhận thức của máy móc. Qua đó, giúp các bác sĩ có thể phản ứng nhanh chóng và điều trị kịp thời cho bệnh nhân.
   * Bằng việc kết hợp khoa học dữ liệu và phân tích dữ liệu, Machine Learning và AI trong tương lai có thể tạo ra bản sao kỹ thuật số của con người dựa trên những dữ liệu chi tiết của họ, qua đó cung cấp hình ảnh toàn diện về bệnh nhân giúp hỗ trợ và giảm rủi ro trong điều trị bệnh.
   * Nghiên cứu cho thấy GDP toàn cầu vào năm 2030 sẽ tăng đáng kể nhờ vào lĩnh vực trí tuệ nhân tạo, biến nó trở thành cơ hội thương mại lớn nhất trong nền kinh tế thay đổi nhanh chóng ngày nay.

## Ý nghĩa đề tài

* Áp dụng Machine Learning & AI vào trong y khoa sẽ giúp các bác sĩ dễ dàng và nhanh chóng hơn trong việc chuẩn đoán bệnh để từ đó kê đơn thuốc đúng và cho từng loại bệnh của các bệnh nhân.
* Giảm thiểu sự sai sót trong chuẩn đoán bệnh.
* Hỗ trợ các bác sĩ từ các quốc gia có nên y khoa chưa phát triển từ đó giúp giảm thiểu rủi ro tử vong của bệnh nhân vì chẩn đoán không chính xác .

VD: Mỗi năm, riêng ở Mỹ, ước tính có khoảng 40.000 đến 80.000 người chết do chẩn đoán sai. (theo BCC News: Lỗi chẩn đoán gây tử vong cho bệnh nhân).

# Mục tiêu

* Nhóm hoàn thành dự án đúng mong đợi:
* Sau tuần học thứ 9 các thành viên trong nhóm đã có thể nắm rõ các thuật toán cơ bản của Machine Learning và chọn được thuật toán phù hợp cho đề tài.
* Sau tuần học thứ 14, nhóm hoàn thành việc thực hiện training model và test, chọn ra được hướng đi có tỉ lệ chính xác cao nhất.
* Các thành viên trong nhóm nắm bắt tiến trình làm việc, dễ dàng hỗ trợ lẫn nhau.
* Từ dữ liệu có sẵn, training ra được bộ model cho chẩn đoán sau này.
* Với hình ảnh test sẽ dự đoán ra được có bệnh hay không có bệnh.
* Kết quả dự đoán đạt độ chính xác cao (trên 80%).
* Có thể áp dụng dự án vào trong thực tế.
* Tương lai sẽ phát triển dự án theo mô hình có khả năng phát hiện và phân loại giai đoạn phát triển của bệnh phổi.

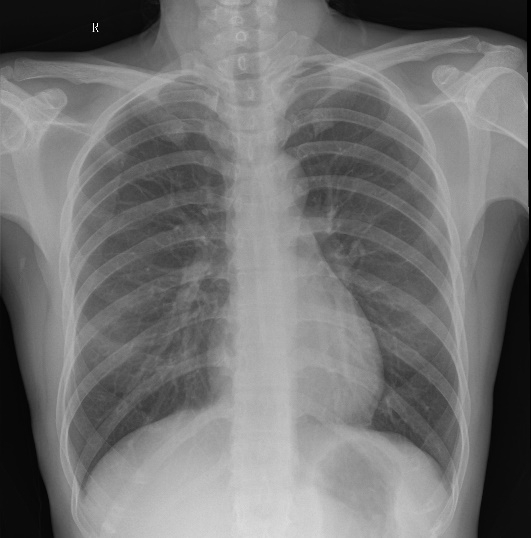
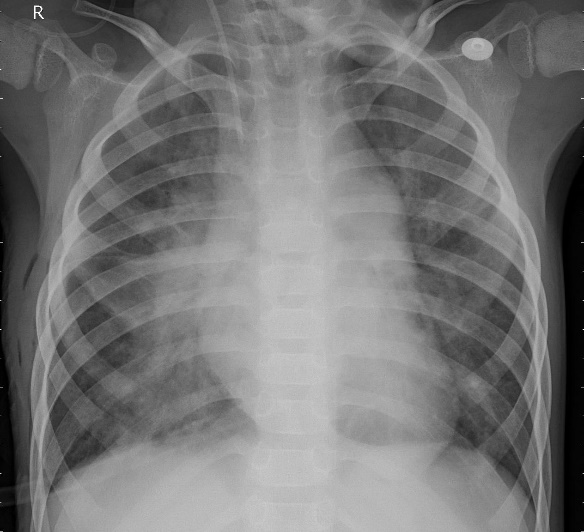
# Quá trình thực hiện

## Đặt vấn đề

* Các thành viên trong nhóm tìm hiểu các khái niệm cơ bản, tìm ra thực trạng và nhu cầu sử dụng cũng như ý nghĩa của đề tài.
  + Sau khi hoàn thành giai đoạn tìm hiểu, các thành viên trong nhóm đã phân tích các vấn đề thực tế có thể áp dụng cho đề tài. Từ đó, hội ý và đưa ra giải pháp cuối cùng để thực hiện đề tài có kết quả tốt nhất có thể.
  + Với tập dữ liệu mới được phân loại thành 2 nhóm: Pneumonia và Normal, nhóm sẽ thực hiện thuật toán Supervised Learning cho đề tài và đáp ứng các mục tiêu, ý nghĩa đã đề ra.

## Chuẩn bị dữ liệu

* + Thông qua Internet, nhóm đã tìm được nguồn dữ liệu hình ảnh X-quang ngực được chọn từ một nhóm nghiên cứu nước ngoài.
  + Tập dữ liệu hình ảnh bao gồm 3 thư mục chính (train, test, val) và chứa 2 thư mục con đã gắn nhãn (Pneumonia, Normal). Tổng cộng có 5863 hình ảnh X-quang (JPEG).
  + Dưới đây là một số hình ảnh trong tập dữ liệu:

Hình 1: Phổi bình thường Hình 2: Phổi bị bệnh

## Chọn thuật toán

* Nhóm thực hiện nghiên cứu, tìm hiểu các thuật toán cơ bản và phổ biến trong chương trình học thông qua các trang web, sự hướng dẫn của giảng viên để phân tích ưu nhược điểm của từng thuật toán và đánh giá mức độ phù hợp cho đề tài.
  + Sau khi tìm hiểu nhiều thuật toán thông qua Internet và trong chương trình học ví dụ như: Support Vector Machine, Convolutional Neural Network, K-means, … và sự hướng dẫn của giảng viên, nhóm đã xác định được thuật toán Convolutional Neural Network phù hợp với đề tài.
  + Lý do chọn thuật toán CNN:
  + Nếu sử dụng các thuật toán cơ bản của Machine Learning thì các dữ liệu phải được tiền xử lý trước khi training.
  + CNN có hỗ trợ tiền xử lý hình ảnh, quá trình training trong thuật toán và đưa ra kết quả.
  + Hiện tại nhóm đang có một lượng dữ liệu hình ảnh lớn, phù hợp với đặc điểm cần nhiều dữ liệu của thuật toán.
  + Giảng viên hướng dẫn đánh giá cao thuật toán về mức độ phù hợp cho đề tài.

## Training model

* + Nhóm thực hiện tìm hiểu và sử dụng các thư viện hỗ trợ cho Python như matplotlib, keras, sklearn…
  + Áp dụng các Layer dùng cho thuật toán CNN:
  + Conv2D: (Convolutional Layers: chứa các layer trong mạng nơ ron tích chập) là convolution layer dùng để lấy feature từ image
  + MaxPool2D: (Pooling Layers: Chứa các layer dùng trong mạng CNN.) dùng để lấy feature nổi bật dùng max) và giúp giảm parameter khi training.
  + Dense: layer này sử dụng như một layer neural network bình thường.
  + Dropout: layer này dùng như regularization cho các layer hạn chế overfiting
  + Softmax: (chọn activation function) dùng trong multi classifier
  + Input: layer này sử dụng input như 1 layer
  + Flatten: dùng để lát phẳng layer để fully connection
  + Activation: dùng để chọn activation.
* Chạy code với các thông số khác nhau về learning rate, tỷ lệ hình trong 2 thư mục (training, test) và cho ra model.

## Kiểm thử

* + Với mỗi model đã được training trong phần trên, nhóm thực hiện import model và kiểm tra tỷ lệ chính xác (%), lưu kết quả kiểm tra với thông số tương ứng.
  + Với kết quả kiểm tra, đưa ra được model có độ chính xác cao nhất và chọn đó làm kết quả cuối cùng.

# Cơ sở lý thuyết

## Xử lý ảnh

* 1. Đặc trưng của ảnh (Feature)

Đặc trưng ảnh là những chi tiết xuất hiện trong ảnh, từ đơn giản như cạnh, hình khối, chữ viết tới phức tạp như mắt, mặt, chó, mèo, bàn, ghế, xe, đèn giao thông...

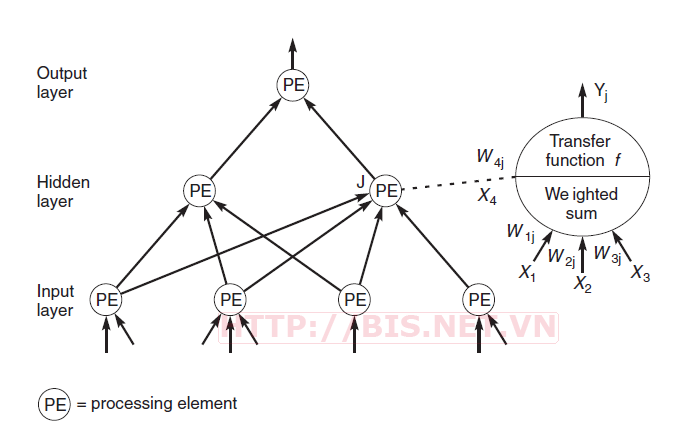
* 1. Bộ lọc ảnh:

Là bộ lọc giúp phát hiện và trích xuất các đặc trừng của ảnh, có thể là bộ lọc góc, cạnh, đường chéo, hình tròn, hình vuông, v.v.

* 1. Trong bài toán của chúng ta thì các bức ảnh là các tấm phim X-ray được chụp của nhiều bệnh nhân và người không bị bệnh. Mỗi hình ảnh có kích thước khác nhau nên muốn xử lý những hình ảnh đó cần trải qua các bước tách các đặt trưng của ảnh để tiện cho việc xử lý. Bước này được gọi là bước tiền xử lý.

## Neural Networks

* 1. Tổng quan
* Mạng nơron nhân tạo, Artificial Neural Network (ANN) là một mô hình xử lý thông tin phỏng theo cách thức xử lý thông tin của các hệ nơron sinh học. Nó được tạo nên từ một số lượng lớn các phần tử (nơron) kết nối với nhau thông qua các liên kết (trọng số liên kết) làm việc như một thể thống nhất để giải quyết một vấn đề cụ thể nào đó.
* Một mạng nơron nhân tạo được cấu hình cho một ứng dụng cụ thể (nhận dạng mẫu, phân loại dữ liệu,... ) thông qua một quá trình học từ tập các mẫu huấn luyện. Về bản chất học chính là quá trình hiệu chỉnh trọng số liên kết giữa các nơron.
  1. Kiến trúc

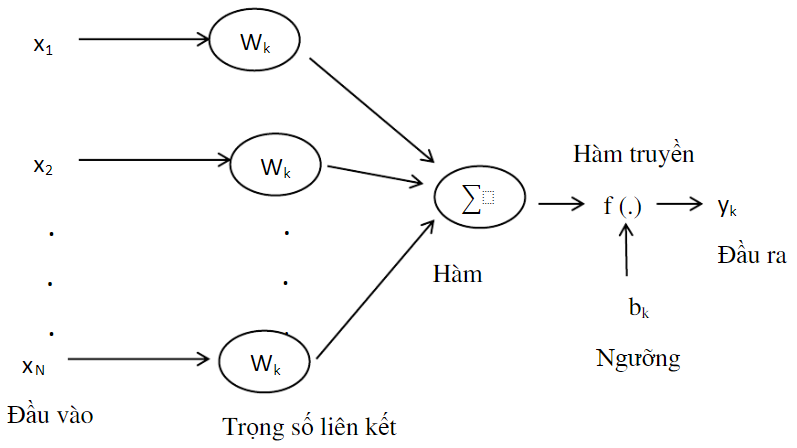


Hình 3: Kiến trúc mạng Neural Networks

* Kiến trúc chung của một NN gồm 3 thành phần đó là Input Layer, Hidden Layer và Output Layer.
  1. Quá trình xử lý thông tin

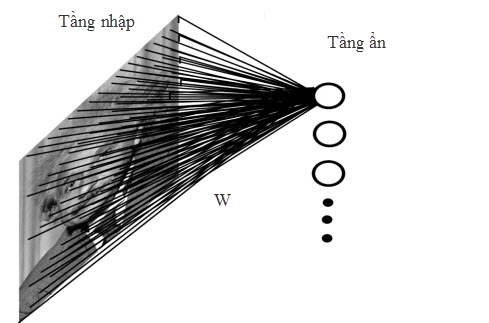


Hình 4: Quá trình xử lý thông tin



Hình 5: Quá trình xử lý thông tin với hàm

* Inputs: Mỗi Input tương ứng với 1 thuộc tính (attribute) của dữ liệu (patterns). Ví dụ như trong ứng dụng của ngân hàng xem xét có chấp nhận cho khách hàng vay tiền hay không thì mỗi Input là một thuộc tính của khách hàng như thu nhập, nghề nghiệp, tuổi, số con,…
* Output: Kết quả của một ANN là một giải pháp cho một vấn đề, ví dụ như với bài toán xem xét chấp nhận cho khách hàng vay tiền hay không thì output là yes (cho vay) hoặc no (không cho vay).
* Connection Weights (Trọng số liên kết) : Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng (độ mạnh) của dữ liệu đầu vào đối với quá trình xử lý thông tin (quá trình chuyển đổi dữ liệu từ Layer này sang layer khác). Quá trình học (Learning Processing) của ANN thực ra là quá trình điều chỉnh các trọng số (Weight) của các input data để có được kết quả mong muốn.
* Bộ tổng (Summing function): Thường dùng để tính tổng của tích các đầu vào với trọng số liên kết của nó
* Ngưỡng (còn gọi là một độ lệch - bias): Ngưỡng này thường được đưa vào như một thành phần của hàm truyền.
* Hàm truyền (Transfer function): Hàm này được dùng để giới hạn phạm vi đầu ra của mỗi nơron. Nó nhận đầu vào là kết quả của hàm tổng và ngưỡng.
* Tuy nhiên, mạng Nơ-ron truyền thống (Neural Network) hoạt động không thực sự hiệu quả với dữ liệu đầu vào là hình ảnh. Nếu coi mỗi điểm ảnh là một thuộc tính (feature), một ảnh RBG kích thước (64×64) có 12288 (=64×64×3) thuộc tính. Nếu kích thước ảnh tăng lên 1000×10000, chúng ta có 3 triệu (3M) thuộc tính cho mỗi ảnh đầu vào. Nếu sử dụng mạng liên kết đầy đủ (fully connected NN) và giả sử lớp thứ 2 có 1000 thành phần (units/ neurons), ma trận trọng số sẽ có kích thước 1000×3M tương đương với 3 tỷ trọng số cần huấn luyện (learning).



Hình 6: ANN (Artificial Neural Network) với cấu trúc Fully – connected

Điều này yêu cầu khối lượng tính toán cực lớn (expensive computational cost) và thường dẫn đến overfitting do không đủ dữ liệu huấn luyện. Nên dẫn đến Convolutional Neural Networks ra đời để khắc phục các vấn đề của mạng Nơ-ron truyền thống (Neural Network).

## Convolutional Neural Networks – CNN: (Mạng nơ-ron tích chập)

* 1. Tổng quan
  + CNN là mạng nơ-ron phổ biến nhất được dùng cho dữ liệu ảnh. Bên cạnh các lớp liên kết đầy đủ (FC layers), CNN còn đi cùng với các lớp ẩn đặc biệc giúp phát hiện và trích xuất những đặc trưng - chi tiết (patterns) xuất hiện trong ảnh gọi là Lớp Tích chập (Convolutional Layers). Chính những lớp tích chập này làm CNN trở nên khác biệt so với mạng nơ-ron truyền thống và hoạt động cực kỳ hiệu quả trong bài toán phân tích ảnh.
  1. Các thành phần cơ bản của mạng CNN

1. Convolution Layer

* Tầng tích chập được dùng để phát hiện và trích xuất đặc trưng - chi tiết của ảnh.



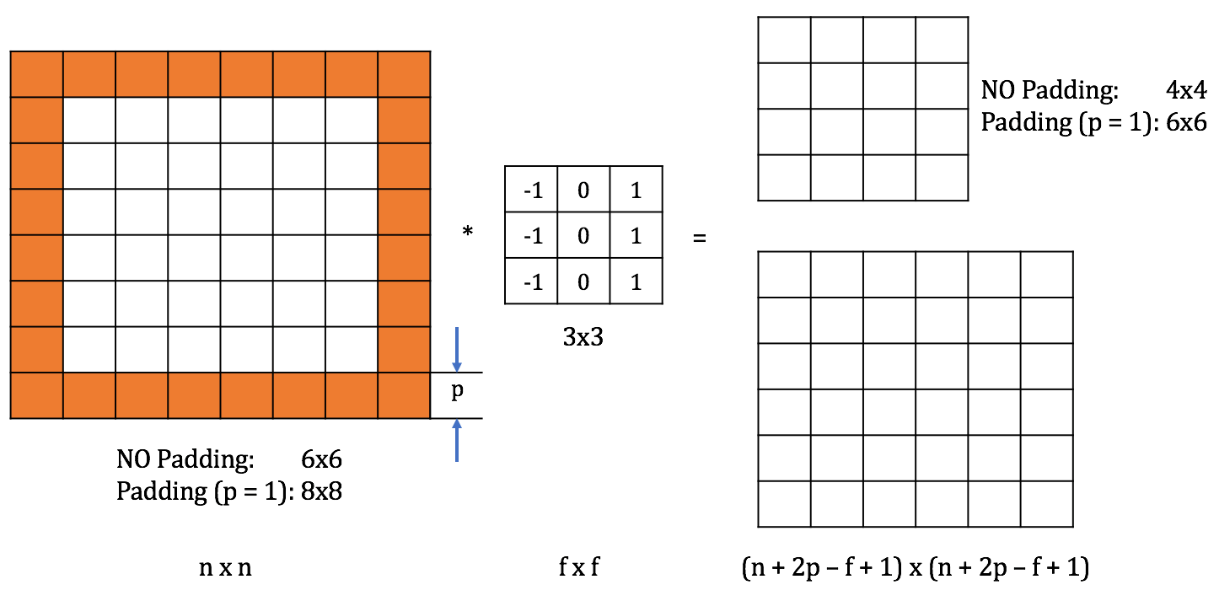
Hình 7: Mô hình phép toán tích chập

* Hình “Minh họa phép toán tích chập” mô tả lý thuyết và cách thức Convolution hoạt động trên một dữ liệu đầu vào được biểu diễn bằng một ma trận hai chiều.
* Ta có thể hình dung phép tính này được thực hiện bằng cách dịch chuyển một cửa sổ mà ta gọi là kernel trên ma trận đầu vào, trong đó kết quả mỗi lần dịch chuyển được tính bằng tổng tích chập (tích của các giá trị giữa 2 ma trận tại vị trí tương ứng), trong hình là giá trị đầu ra khi dịch chuyển kenel kích thước 2x2 trên toàn bộ ma trận kích thước 3x4 (Ảnh minh họa).
* Sliding Window hay còn gọi là kernel, filter hoặc feature detect là một ma trận có kích thước nhỏ thường là 3x3 hoặc 5x5 đối với đầu vào là một tra trận có kích thước lớn, dùng để trích xuất các đặc trưng của ma trận đầu vào.
* Khi được áp dụng phép tính Convolution vào xử lý ảnh người ta thấy rằng Convolution sẽ giúp biến đổi các thông tin đầu vào thành các yếu tố đặc trưng (nó tương ứng như bộ phát hiện – detector features về cạnh, hướng, đốm màu …).
* Giống như các tầng ẩn khác, tầng Convolution (tích chập) lấy dữ liệu đầu vào, thực hiện các phép chuyển đổi để tạo ra dữ liệu đầu vào cho lớp kế tiếp (đầu ra của tầng này là đầu vào của tầng sau). Phép biến đổi được sử dụng là phép tính tích chập. Mỗi lớp tích chập chứa một hoặc nhiều bộ lọc - bộ phát hiện đặc trưng (filter - feature detector) cho phép phát hiện và trích xuất những đặc trưng khác nhau của ảnh.
* Bộ lọc ở lớp tích chập càng sâu thì phát hiện các đặc trừng càng phức tạp.
* Trong mạng CNN, những lớp tích chập đầu tiên sử dụng bộ lọc hình học (geometric filters) để phát hiện những đặc trưng đơn giản như cạnh ngang, dọc, chéo của bức ảnh. Những lớp tích chập sau đó được dùng để phát hiện đối tượng nhỏ, bán hoàn chỉnh như mắt, mũi, tóc, v.v. Những lớp tích chập sâu nhất dùng để phát hiện đối tượng hoàn hỉnh như: chó, mèo, chim, ô tô, đèn giao thông, v.v.



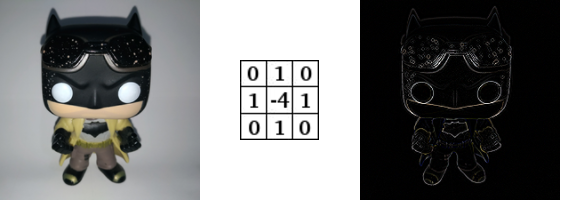
Hình 8: Hình ảnh sau khi lọc

* Tại sao phải dùng padding trong một số tầng Convolutional:
* Lấy ví dụ với ma trận đầu vào kích thước 6×6. Nếu ta nhân chập với bộ lọc kích thước 3×3, kết quả thu được là một ma trận đầu ra kích thước 4×4 vì chỉ có 4×4 vị trí trên ma trận đầu vào để đặt ma trận lọc. Tổng quát hoá, nếu ta nhân chập ma trận đầu vào kích thước n×n với bộ lọc kích thước f×f, ta thu được kết quả là một ma trận kích thước (n−f+1)×(n−f+1). Mỗi một lần áp dụng phép nhân chập, kích thước của ảnh bị giảm xuống, và vì thế chúng ta chỉ có thể thực hiện nó một vài lần trước khi ảnh trở nên quá nhỏ.
* Điểm ảnh ở khoảng trung tâm của ma trận đầu vào được bao phủ bởi rất nhiều vùng 3×3 nghĩa là được sử dụng để tính nhiều giá trị đầu ra, trong khi những điểm ảnh ở góc hoặc cạnh chỉ được sử dụng 1 hoặc 2 lần vì chỉ bị bao phủ bởi 1 hoặc 2 vùng 3×3. Vì thế chúng ta đánh mất rất nhiều thông tin (có thể quan trọng) tại các vùng gần cạnh của ảnh.



Hình 9: Ma trận đầu vào được bao quanh bởi đường viền phụ kích thước p (giá trị 0)

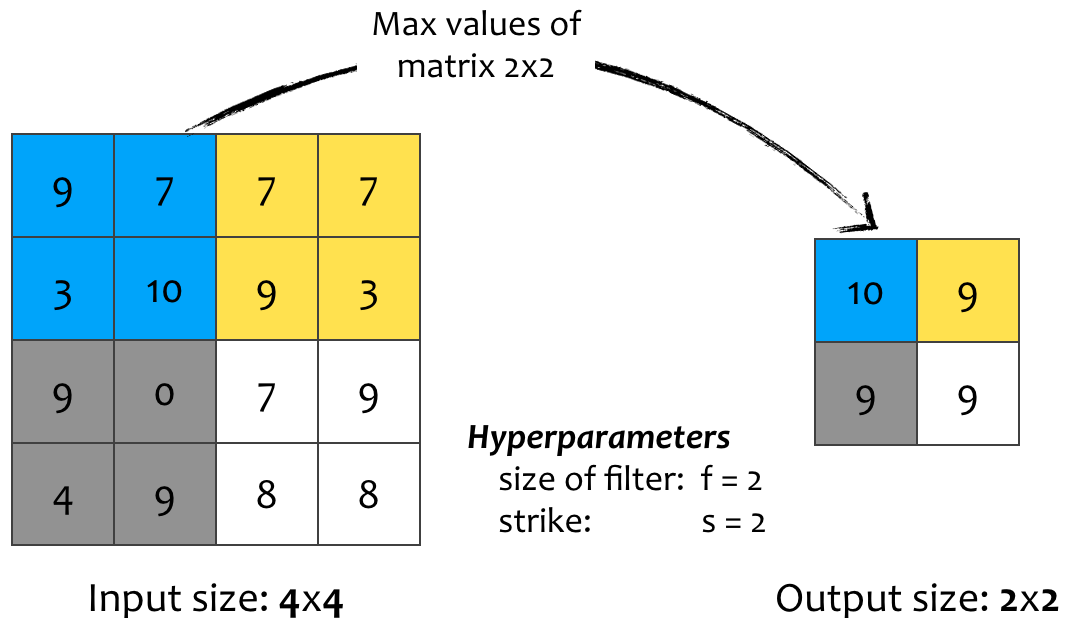
* Để khắc phục hai nhược điểm trên, một đường viền phụ (padding) được thêm vào xung quanh ma trận đầu. Việc thêm đường viền phụ làm tăng kích thước của ma trận đầu vào, dẫn tới tăng kích thước ma trận đầu ra. Từ đó độ chênh lệch giữa ma trận đầu ra với ma trận đầu vào gốc giảm. Những ô nằm trên cạnh/ góc của ma trận đầu vào gốc cũng lùi sâu vào bên trong hơn, dẫn tới được sử dụng nhiều hơn trong việc tính toán ma trận đầu ra, tránh được việc mất mát thông tin.
* Trong hình 5, ma trận đầu vào kích thước 6×6 được thêm vào đường viền phụ kích thước 1 (p=1), trở thành ma trận 8×8. Khi nhân chập ma trận này với bộ lọc 3×3, chúng ta thu được ma trận đầu ra 6×6. Kích thước của ma trận đầu vào (gốc) được duy trì. Những điểm ảnh nằm ở cạnh của ma trận đầu vào gốc được sử dụng nhiều lần hơn (4 lần với những điểm ảnh ở góc).
* Theo quy ước các ô trên padding có giá trị bằng 0.
* Theo quy ước, kích thước bộ lọc f là số lẻ vì hai lý do chính sau:
  + Nếu f là số chẵn, chúng ta phải thêm vào bên trái của ma trận đầu vào nhiều hơn bên phải (hoặc ngược lại), việc này dẫn tới hệ đầu vào không đối xứng (**asymetric**).
  + Nếu f là số lẻ, ma trận đầu vào có một điểm ảnh ở trung tâm. Trong lĩnh vực thị giác máy tính, việc có một nhân tố khác biệt (distinguisher) - một điểm đại diện cho vị trí của bộ lọc thường mang lại hiệu năng cao cho bài toán.



Hình 10: Một ảnh sau khi được Convolution với Kernel = [0, 1, 0, 1, -4, 1, 0, 1, 0]

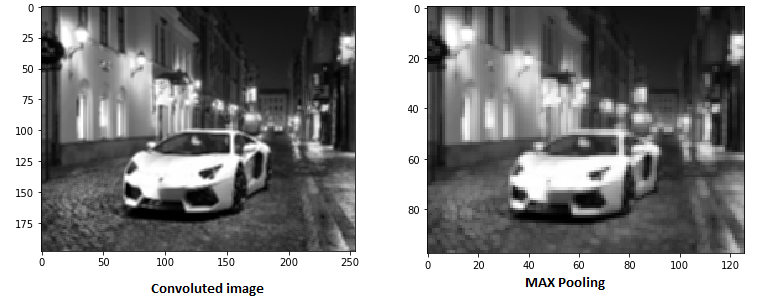
1. Pooling

* Tầng Pooling được sử dụng trong CNN để giảm kích thước đầu vào, tăng tốc độ tính toán và hiệu năng trong việc phát hiện các đặc trưng. Có nhiều hướng Pooling được sử dung, trong đó phổ biến nhất là pooling theo giá trị cực đại (max pooling) .
* Pooling layer cung cấp tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling).
* Pooling theo giá trị cực đại (Max pooling):
* Nếu một đặc trưng được phát hiện ở một vùng nào đó bị bao phủ bởi bộ lọc, giá trị cao nhất trong vùng sẽ được giữ lại tuy nhiên chưa ai giải thích được tại sao cách tiếp cận này lại hoạt động tốt trong thực nghiệm.



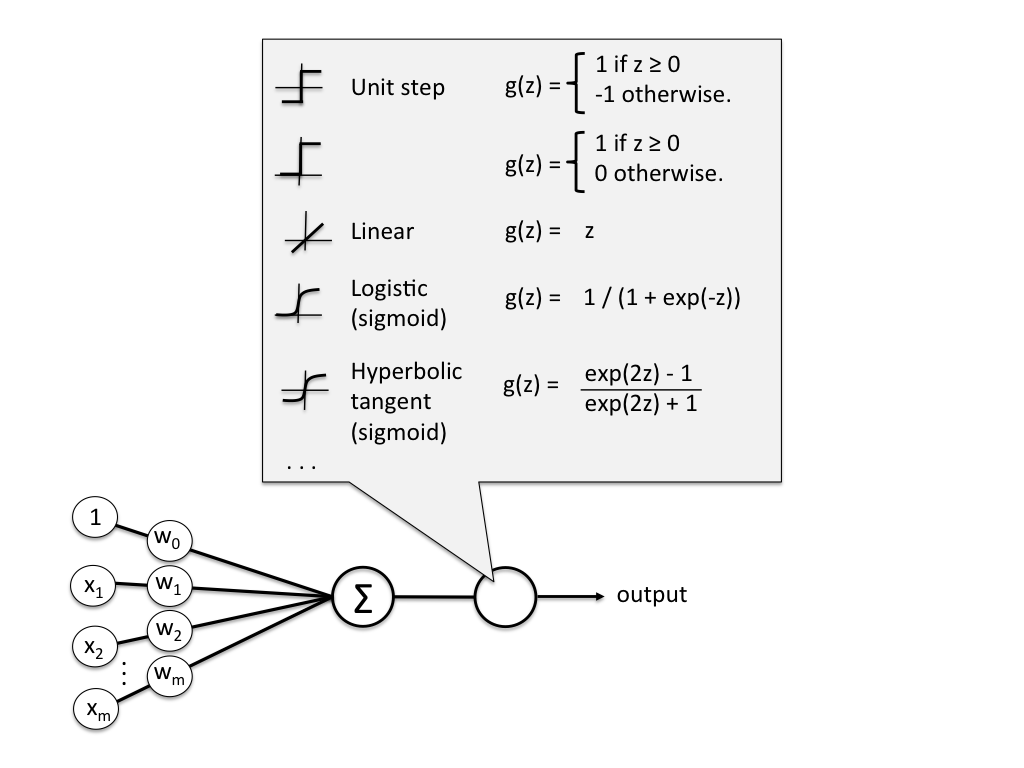
Hình 11: Ví dụ pooling theo giá trị cực đại

Bộ lọc kích thước 2x2 trượt trên ma trận đầu vào 2 hàng/cột trong mỗi bước nhảy (s = 2) và chia nó thành những vùng khác nhau. Mỗi ô trong ma trận đầu ra lấy giá trị lớn nhất của vùng tương ứng.



Hình 12: So sánh sau khi Max Pooling từ ảnh đã được Convolution

1. Activation function (Hàm kích hoạt)

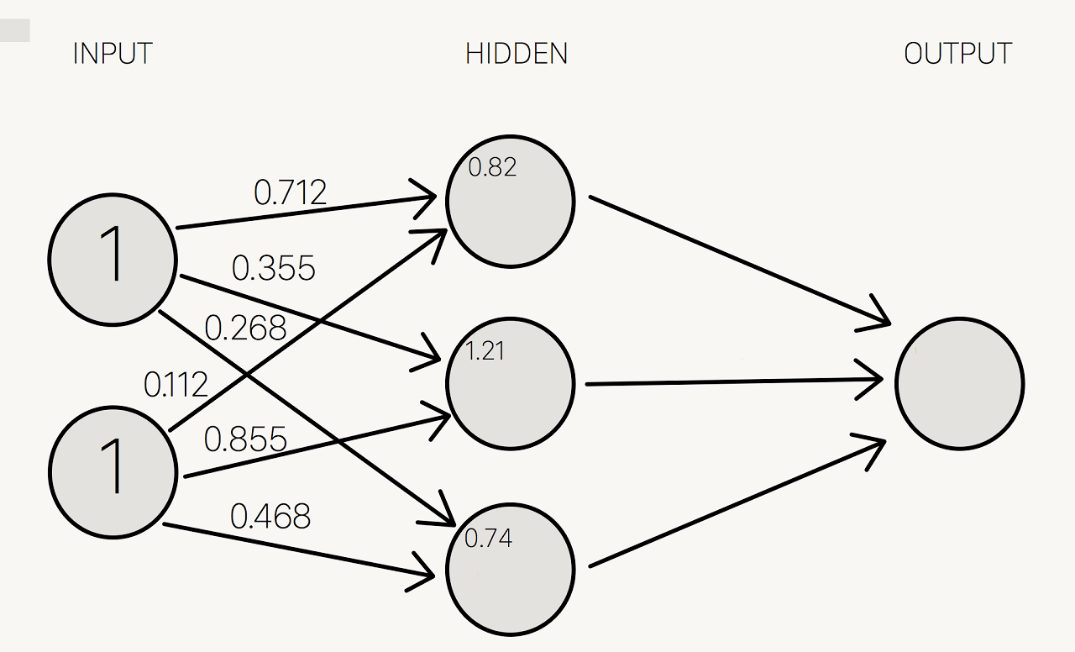


Hình 13: Hàm kích hoạt

Về cơ bản, Activation Function là một phép biển đổi tuyến tính. Nếu tất cả các neural được tổng hợp bởi các phép biến đổi tuyến tính thì một mạng neural đều có thể đưa về dưới dạng một hàm tuyến tính. Khi đó sẽ đưa các bài toán về Logistic Regression. Do đó tại mỗi neural cần có một hàm truyền dưới dạng phi tuyến. Có nhiều dạng hàm phi tuyến được sử dụng trong quá trình này như tanh, sigmoid function, hay ReLUs.

Tuy nhiên, các nghiên cứu gần đây chứng minh được việc sử dụng hàm ReLu (Rectified Linear Unit) cho kết quả tốt hơn ở các khía cạnh:

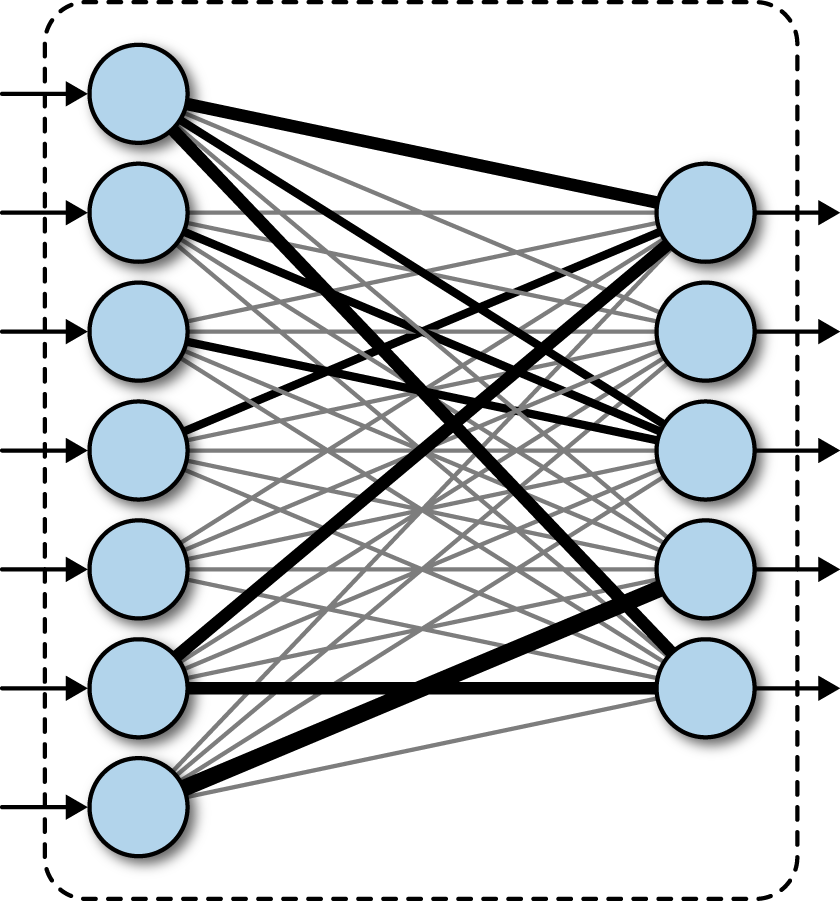
* Tính toán đơn giản.
* Tạo ra tính thưa (sparsity) ở các neural ẩn. Ví dụ như sau bước khởi tạo ngẫu nhiên các trọng số, khoảng 50\% các neural ẩn được kích hoạt (có giá trị lớn hơn 0).
* Quá trình huấn luyện nhanh hơn ngay cả khi không phải trải qua bước tiền huấn luyện.
* Trong bài toán này chúng ta sử dụng hàm ReLU để làm hàm Activation.



Hình 14: Một Neural Network đơn giản áp dụng ReLU

1. Fully – Connected

Là cách kết nối các neural ở hai tầng với nhau trong đó tầng sau kết nối đẩy đủ với các neural ở tầng trước nó. Đây cũng là dạng kết nối thường thấy ở ANN (Artificial Neural Network), trong CNN tầng này thường được sử dụng ở các tầng phía cuối của kiến trúc mạng.



Hình 15: Cách hoạt động của Fully-Connected

Tất cả các Node ở phía sau đều được liên kết với các Node phía trước

* 1. Cấu trúc mạng CNN và cách hoạt động

CNN có kiến trúc được hình thành từ các thành phần cơ bản bao gồm Convolution (CONV), Pooling (POOL), ReLU, Fully-connected (FC) về mặt xây dựng kiến trúc tổng quát CNN được mô tả như sau (dấu mũi tên thể hiện thứ tự sắp xếp các tầng từ trước đến sau):



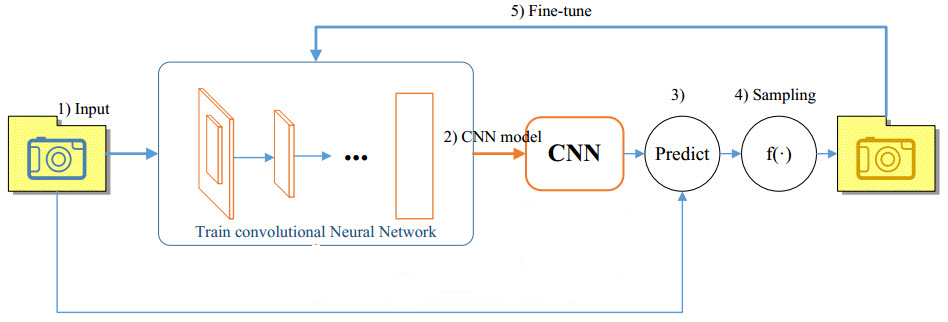
Hình 16: Quá trình hoạt động của mạng CNN

Quá trình hoạt động của mạng CNN:

* + Convolution + ReLU tức là trong kiến trúc này sau khi Convolution thì sẽ kết hợp ReLU, trong CNN kiến trúc lớp này có thể lặp N lần.
  + Pooling là tầng Pooling cho người thiết kế quyết định có thể có hoặc không.
  + Tập hợp các lớp Convolution + ReLU và Pooling được gọi là giai đoạn Feature Learning.
  + Sau đó tập hợp các Convolution Layer chồng lên nhau và sử dụng các nonlinear activation function như ReLU để kích hoạt các trọng số trong các node.
  + Mỗi một lớp sau khi thông qua các hàm kích hoạt sẽ tạo ra các thông tin trừu tượng hơn cho các lớp tiếp theo.
  + Các layer liên kết được với nhau thông qua cơ chế convolution. Layer tiếp theo là kết quả convolution từ layer trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Nghĩa là mỗi nơ-ron ở layer tiếp theo sinh ra từ filter áp đặt lên một vùng ảnh cục bộ của nơ-ron layer trước đó.
  + Sau khi kết thúc quá trình Feature Learning thì có 2 lớp Fatten và Fully Connected sẽ tạo ra các liên kết đầy đủ như một mạng Neurual Network bình thường. Cuối cùng sẽ cho ra kết quá mỗi hình được xử lí thuộc nhóm nào.
* Trong mô hình CNN có 2 khía cạnh cần quan tâm là tính bất biến (Location Invariance) và tính kết hợp (Compositionality):
* Tính bất biến: Với cùng một đối tượng, nếu đối tượng này được chiếu theo các gốc độ khác nhau (translation, rotation, scaling) thì độ chính xác của thuật toán sẽ bị ảnh hưởng đáng kể. Pooling layer sẽ cho bạn tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling).
* Tính kết hợp: cho ta các cấp độ biểu diễn thông tin từ mức độ thấp đến mức độ cao và trừu tượng hơn thông qua convolution từ các filter.
  + Trong suốt quá trình huấn luyện, CNNs sẽ tự động học được các thông số.

# Quá trình triển khai xử lý bài toán

## Flow chart



Hình 17: Quá trình triển khai xử lý bài toán

Quá trình:

Bước 1: Input bộ data gồm bộ data training và bộ data testing

Bước 2: Từ bộ training trải qua quá trình xử lý CNN tạo ra được bộ Model

Bước 3: Sử dụng bộ Model tạo được so sánh với bộ testing

Bước 4: Đánh giá mô hình

Bước 5: Nếu kết quả đánh giá đáp ứng nhu cầu thì dừng lại. Ngược lại thì quay lại bước 2.

## Xử lý dữ liệu

Quá trình xử lý ảnh dữ liệu đối với một mạng CNN gồm có 4 bước:

* + Bước 1: Dùng Convolutional để trích xuất đặt trưng của ảnh.
  + Bước 2: Dùng Pooling để giảm số chiều ma trận trong khối ma trận.

(Bước 1 và Bước 2 có thể lặp lại nhiều lần tùy theo độ phức tạp và đặt trưng của ảnh, riêng Bước 2 có thể có hoặc không)

* + Bước 3: Lát phẳng khối ma trận đặt trưng (Flatten)
  + Bước 4: Fully Connected

Đối với quá trình xử lý dữ liệu Dự Án có 15 layer được thực hiện theo thứ tự và chức năng như sau:

* + Layer 1: Là lớp Convolutional đầu tiên với 32 filters, mỗi filter là một ma trận có kích thước 3x3, thông qua thư viện Keras ta có được cách sinh các trọng số trên mỗi filter thông qua phương thức glorot-uniform với việc lấy phân phối đồng đều các giới hạn âm và giới hạn dương với:

Trong đó:

* + - fan\_in là số đơn vị đầu vào có node.
    - fan\_out là số đơn vị đầu ra đạt được.

Sau khi một filter đã trược hết trên ma trận input thì ta sẽ được một ma trận mới với kích thước theo công thức:

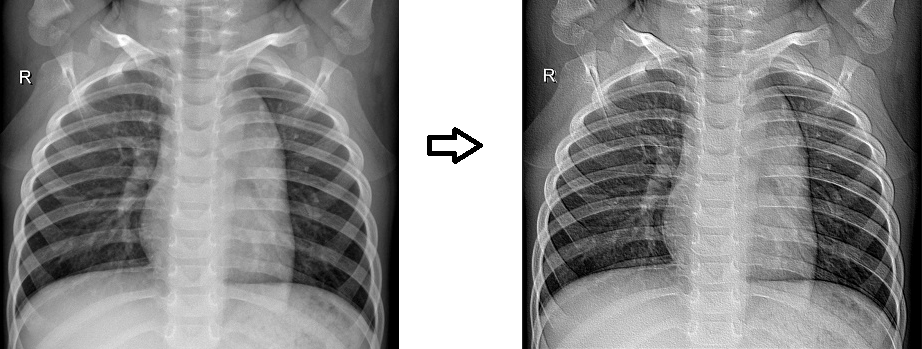
( n – f + 1 + 2p) x ( n – f + 1 + 2p)

Trong đó:

* + - n là kích thước của ma trận ảnh hay ma trận đầu vào lớp Convolutional.
    - f là kích thước của filter trược.
    - p là kích thước padding thông thường (p=1).

Như vậy sau khi hoàng thành xong Layer 1 kết quả đạt được là một khối ma trận gồm 32 ma trận, với mỗi ma trận có kích thước là 64x64 (từ ảnh bản đầu có kích thước 64x64 có số chanel là 1).

Số tham số của Layer này là (3 x 3 + 1) x 32 = 320 tham số. Còn nếu dùng Fully Connected thì số tham số sẽ là 64 x 64 = 4096 tham số.

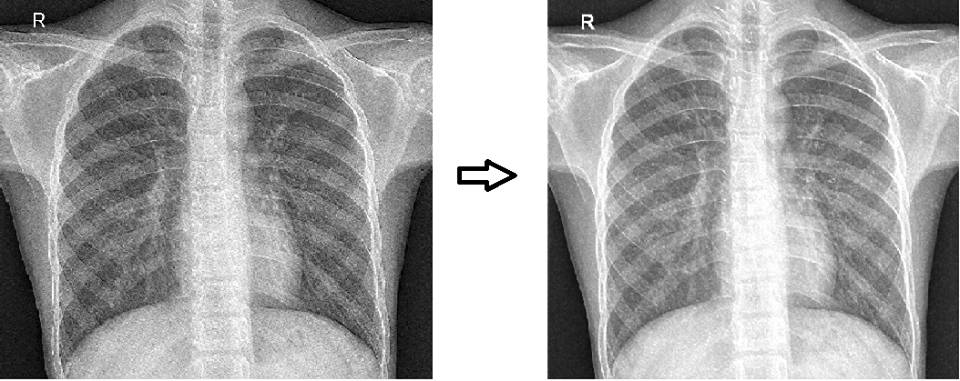


Hình 18: Một ảnh phổi sau khi được Convolution với Kernel = [-2,-1,0,-1,1,1,0,1,2]

* + Layer 2: Là lớp Convolutional thứ 2 với 32 filters mỗi filter có kích thước 3 x 3 và không có padding, được dùng trên khối ma trận sau khi qua Layer 1 với 32 ma trận có kích thước 64 x 64. Với các bước tính toán thì sau khi các filters đã trược hết trên khối ma trận ta sẽ thu được một khối ma trận mới với 1024 ma trận có kích thước 62 x 62.
  + Layer 3: Batch Normaliztion sẽ giúp làm giảm sự thay đổi của các lớp ẩn thông qua việc chuẩn hóa khối ma trận có được từ Layer 2. Với 2 thuật toán tham gia vào Layer này là:
    - Thuật toán đảm bảo sự biến đổi ban đầu của một lớp với giá trị đã dịch chuyển và chuẩn hóa.
    - Thuật toán tạo tổng thể của mạng được chuẩn hóa hàng loạt.
  + Layer 4: Max Pooling giúp làm giảm kích thước của các ma trận trong khối 1024 ma trận (62 x 62) thông qua đó sẽ lấy ra được các đặt trưng nỗi bật nhất trên từng vùng mà ma trận Max Pooling kích thước 2 x 2 nhảy qua. Trong ma trận Max Pooling (2 x 2) ta có 2 tham số cần lưu ý:
    - Bước nhảy s = 2
    - Kích thước f x 2 = 2 x 2
    - Như vậy sau khi ma trận Max Pooling đã nhảy qua toàn bộ ma trận đầu vào thì kích thước của ma trận đó sẽ giảm còn (n + 2p - f) / s + 1 (với p là padding nếu có).

Vậy sau khi qua Layer 4 ta sẽ có được 1 khối ma trận gôm 1024 ma trận với kích thước 31 x 31.

Thông qua lớp này sẽ đảm bảo tính nhất biến của các đặt trưng nỗi bật khi có sự thay đổi: rotation, rolling, scaling.



Hình 19: So sánh sau khi Max Pooling từ ảnh đã được Convolution

* + Layer 5: Dùng Dropout trên khối ma trận để giả định các phần tử nào không quan trọng ta có thể bỏ qua nhằm hạn chết overfitting với tỷ lệ là 0.25.
  + Layer 6: Tiếp tực là một Convolutional Layer có padding với 64 filters với mỗi filter có kích thước 3 x 3. Vậy sau khi trược qua hết 1024 ma trận ta sẽ thu được một khối mới với số ma trận là: 64 x 1024 = 65536 ma trận kích thước 31 x 31.
  + Layer 7: Tiếp tực là một Convolutional Layer có padding với 64 filters với mỗi filter có kích thước 3x3. Vậy sau khi trược qua hết 65536 ma trận ta sẽ thu được một khối mới với số ma trận là: 64 x 65536 = 4194304 ma trận kích thước 31 x 31.
  + Layer 8: Batch Normaliztion tương tự Layer 3.
  + Layer 9: Max Pooling tiếp tục giảm kích thước, nhưng đối với ma trận kích thước lẻ 31 x 31 thì Keras sẽ tự điều phối tính toán kích thước.
  + Layer 10: Dùng Dropout tương tự Layer 5.
  + Layer 11: Flatten: Vì trong quá trình tạo các lớp ẩn thì thông qua các lớp ẩn các ma trận bị chồng lên nhau tạo ra một khối ma trận. Với ảnh đầu vào là ảnh xám 2D để đảm bảo bảo toàn thông tin của các đặt trưng trên ảnh trước khi đến đầu ra ta cần lát phẳng khối ma trận để đảm bảo Fully Connected có thông tin trên các Node là đặc trưng và chuẩn nhất.
  + Từ Layer 12 đến Layer 15 là quá trình Fully Connected: Trong đó có 2 lần Fully Connected:
    - Lần đầu là ở Layer 12: Với hàm kích hoạt là Relu để giảm chi phí tính toán.
    - Lần hai là ở Layer 15: Với hàm softmax để dự đoán độ chính xác của ảnh trong phân lớp.
  + Layer 15: Với bài toán Dự đoán ảnh bệnh phổi cho 2 đầu ra có bệnh - không có bệnh nếu ta dùng hàm softmax ta có thể dự đoán phần trăm chính xác của đầu ra trong 2 trường hợp có tổng là 100%. Vậy :
    - Có bệnh chiếm tỷ lệ n%
    - Không bệnh chiếm tỷ lệ 100% - n%
  + Tại sao ta phải dùng Fully Connected?
    - Kết quả thu được từ quá trình trích xuất các đặt trưng trong các lớp ẩn thông qua Convolutional là các đặt trưng lớn của ảnh. Và thông qua quá trình lát phẳng ra thì các ma trận trong khối là rời rạc nhau, nên cần phải có fully connected để đảm bảo tổng hợp lại các đặt trưng của ảnh.

Quá trình Convolutional ta sử dụng hàm Non-linear Activation là Relu vì nó cho kết quả tốt hơn trên nhiều khía cạnh khác nhau:

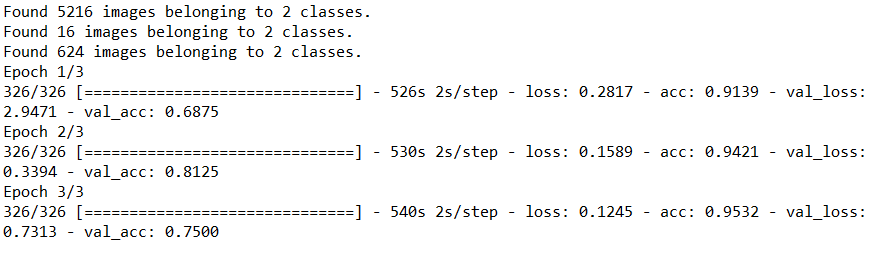
* + - Tính toán đơn giản.
    - Tính toán thưa ở các Neural ẩn.
    - Quá trình huấn luyện nhanh hơn.

## Huấn luyện

Để huấn luyện ảnh đầu tiên ta cần thực hiện xám hóa ảnh. Và đảm bảo kích thước của ảnh là 64 x 64:

* + Với bộ dữ liệu huấn luyện gồm 5780 ảnh y kho về bệnh phổi, tốc độ học lần lượt là 0,001; 0.0001 và sử dụng hàm Adam để biên tập lại model sau khi build. Thì mong muốn trên kết quả sẽ là một tỷ lệ tốt.
  + Bên cạnh đó ta còn thay đổi tỉ lệ giữa train và test để tìm ra tỉ lệ thích hợp nhất và trong bài toàn này thì tỉ lể lý tưởng nhất là 96:1.

Tiếp theo sử dụng các hàm train có sẵn trong thư viện Keras để hỗ trợ việc training, với đầu vào là 2 đường dẫn, mỗi đường dẫn chỉ 1 thư mục mà trong mỗi thư mục đều có chứa 2 tập dữ liệu gồm NORMAL và PE. Với qui định mặc định Label 0 là thư mục tập các ảnh phổi không bị bệnh và Label 1 là thư mục tập các ảnh phổi bị bệnh. Từ đó trong lúc Train thư viện sẽ tự biết và phân nhóm các hình ảnh. Từ đó cho ra các con số thể hiện sự chính xác trong lúc training.



Hình 20: Sau khi training bộ dữ liệu với Learning rate = 0,0001 và tỉ lệ hình 9:1

# Kết quả và đánh giá:

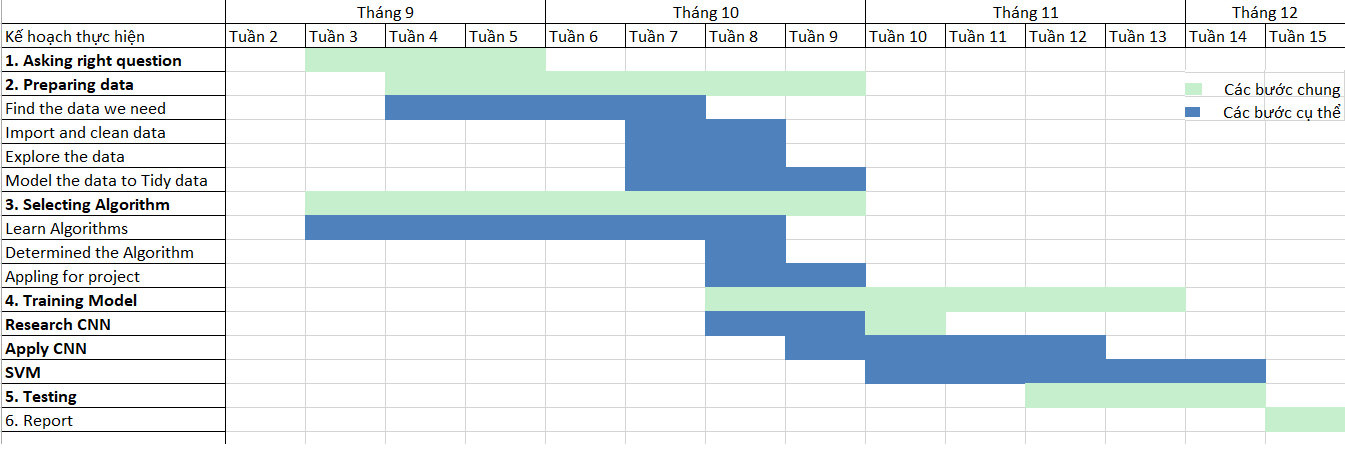
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Learning rate | Số hình ảnh train | Số hình ảnh test | Tỷ lệ hình ảnh | Độ chính xác |
| MODEL\_1 | 0,001 | 5216 | 624 | 9:1 | 68,75% |
| MODEL\_2 | 0,0001 | 5216 | 624 | 9:1 | 75% |
| MODEL\_3 | 0,001 | 5780 | 60 | 96:1 | 83.33% |
| MODEL\_4 | 0,0001 | 5780 | 60 | 96:1 | 91.67% |

Bảng 1: Thống kê độ chính xác của model sau nhiều lần thay đổi thông số

* Với bảng kết quả, ta có thể thấy việc thay đổi learning rate phù hợp và tăng số lượng hình ảnh training lên cho thuật toán CNN là rất cần thiết. Ngược lại, nếu ta chọn learning rate không phù hợp hoặc số lượng hình ảnh training thấp thì sẽ cho kết quả có độ chính xác không cao.

Sau nhiều thay đổi về thông số learning rate và tỷ lệ hình ảnh train/test, nhóm đã quyết định sử dụng MODEL\_4 để cho ra kết quả tốt nhất.

# Kế hoạch thực hiện

****

Hình 21: Bảng kế hoạch thực hiện

# Tài liệu tham khảo

1. [https://machinelearningcoban.com](https://machinelearningcoban.com/)
2. <https://www.coursera.org/learn/machine-learning> (Andrew Ng)
3. Machine learning in Python Essential Techniques For Predictive Analysis (Book)
4. [https://www.tensorflow.org](https://www.tensorflow.org/)
5. https://www.kaggle.com
6. <https://viblo.asia>
7. <https://app.pluralsight.com/library/courses/python-understanding-machine-learning>

Đường dẫn dữ liệu https://drive.google.com/drive/folders/1sLhEOjz9m1kqCs\_wyXVZpAqw4OB06WEg?usp=sharing