**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**- - - 🙞🕮🙜 - - -**

**Tiểu luận chuyên ngành**

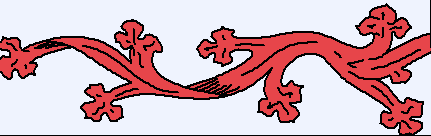
**Ngành: CÔNG NGHỆ THÔNG TIN**

**TÌM HIỂU HỌC SÂU CHO BÀI TOÁN PHÂN LOẠI NGHỀ NGHIỆP**

BOOKS

**GVHD:TS. Nguyễn Thiên Bảo**

**SVTH:**Nguyễn Huy Quang - 15110107

SHAPE056

Tp. Hồ Chí Minh, tháng 12 năm 2018

 CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

\*\*\*

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020

# NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH

Họ và tên sinh viên: Nguyễn Huy Quang MSSV: 15110107  
Chuyên ngành: Công nghệ phần mềm Lớp: 15110CL4  
Giảng viên hướng dẫn: T.S Nguyễn Thiên Bảo  
Ngày nhận đề tài: 14/09/2020 Ngày nộp đề tài: 23/12/2020

1. Tên đề tài: Tìm hiểu học sâu cho bài toán phân loại ảnh nghề nghiệp
2. Các số liệu, tài liệu ban đầu: Data training từ GVHD và Google Image, tài liệu nghiên cứu Deep Learning, CNN từ GVHD và tự tìm kiếm.
3. Nội dung thực hiện đề tài:

* Xây dựng các mô hình phỏng đoán nghề nghiệp dựa trên hình ảnh, sử dụng CNN.
* Xây dựng ứng dụng trực quan phân loại nghề nghiệp dựa trên mô hình có tỉ lệ cao nhất.

1. Sản phẩm: Một API và một ứng dụng phân loại nghề nghiệp dựa trên models có tỉ lệ cao nhất.

TRƯỞNG NGÀNH GIẢNG VIÊN HƯỚNGDẪN

 CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

\*\*\*

# PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên sinh viên: Nguyễn Huy Quang MSSV: 15110107

Chuyên ngành: Công nghệ phần mềm

Tên đề tài: Tìm hiểu học sâu cho bài toán phân loại ảnh nghề nghiệp

Họ và tên Giáo viên hướng dẫn: T.S Nguyễn Thiên Bảo

**NHẬN XÉT**

1. Về nội dung đề tài & khối lượng thực hiện:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Ưuđiểm:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Khuyết điểm:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Đề nghị cho bảo vệ hay không?

..................................................................................................................................

1. Đánh giá loại:

..................................................................................................................................

6. Điểm:..........................(Bằng chữ: )

..................................................................................................................................

Tp. Hồ ChíMinh, ngày tháng năm 20

Giáo viên hướng dẫn  
 (*Ký & ghi rõ họ tên)*

 CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

\*\*\*

# PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên sinh viên: Nguyễn Huy Quang MSSV: 15110107

Chuyên ngành: Công nghệ phần mềm

Tên đề tài: Tìm hiểu học sâu cho bài toán phân loại ảnh nghề nghiệp

Họ và tên Giáo viên phản biện: Th.S Nguyễn Trần Thi Văn

**NHẬN XÉT**

1. Về nội dung đề tài & khối lượng thực hiện:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Ưuđiểm:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Khuyết điểm:

..................................................................................................................................

..................................................................................................................................

..................................................................................................................................

1. Đề nghị cho bảo vệ hay không?

..................................................................................................................................

1. Đánh giá loại:

..................................................................................................................................

6. Điểm:..........................(Bằng chữ: )

..................................................................................................................................

Tp. Hồ ChíMinh, ngày tháng năm 20

Giáo viên phản biện  
 (*Ký & ghi rõ họ tên)*

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến GVHD T.S Nguyễn Thiên Bảo. Trong quá trình học tập và tìm hiểu môn Tiểu luận chuyên ngành, chúng em đã nhận được sự quan tâm giúp đỡ, hướng dẫn rất tận tình, tâm huyết của thầy. Thầy đã giúp chúng em tích lũy thêm nhiều kiến thức để có cái nhìn sâu sắc và hoàn thiện hơn trong đồ án.Từ những kiến thức thầy đã truyền đạt, chúng em đã dần thấm nhuần được những lý thuyết môn Tiểu luận chuyên ngành.

Có lẽ kiến thức là vô hạn mà sự tiếp nhận kiến thức của bản thân mỗi người luôn tồn tại những hạn chết nhất định. Do đó, trong quá trình hoàn thành bài tiểu luận này, chắc chắn không tránh khỏi những thiếu sót. Bản thân chúng em rất mong nhận được những góp ý đến từ thầy để bài tiểu luận của chúng em dần được hoàn thiện hơn.

Tp. Hồ Chí Minh, ngày 23 tháng 12 năm 2020  
 Sinh viên thực hiện

Nguyễn Huy Quang

**MỤC LỤC**

[NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH 2](#_Toc59612144)

[PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 3](#_Toc59612145)

[PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 4](#_Toc59612146)

[LỜI CẢM ƠN 5](#_Toc59612147)

[Chương I: Tổng quan bài toán phân loại nghề nghiệp 11](#_Toc59612148)

[1. Lý do lựa chọn đề tài 12](#_Toc59612149)

[2. Ý nghĩa đề tài 13](#_Toc59612150)

[3. Kết quả mong đợi 13](#_Toc59612151)

[Chương II: Tổng quan về học sâu 14](#_Toc59612152)

[1. Tổng quan học máy 14](#_Toc59612153)

[2. Tổng quan học sâu 16](#_Toc59612154)

[3. Các ứng dụng của học sâu 17](#_Toc59612155)

[Chương III: Mô hình CNN 19](#_Toc59612156)

[1. Tổng quan về CNN 19](#_Toc59612157)

[2. Đặc điểm của CNN 20](#_Toc59612158)

[a. Áp dụng Neural Network 20](#_Toc59612159)

[b. Ưu và nhược điểm của CNN 21](#_Toc59612160)

[3. Cách hoạt động của CNN 22](#_Toc59612161)

[4. Các thành phần cơ bản của mạng CNN 24](#_Toc59612162)

[a. Convolution 24](#_Toc59612163)

[b. Max pooling 28](#_Toc59612164)

[c. Fully – connected 29](#_Toc59612165)

[5. Hàm kích hoạt (activation function) 30](#_Toc59612166)

[a. Sigmoid function 30](#_Toc59612167)

[b. Softmax 30](#_Toc59612168)

[c. Rectified linear unit (ReLU) 31](#_Toc59612169)

[Chương IV: Áp dụng học sâu vào bài toán 32](#_Toc59612170)

[1. Kế hoạch 32](#_Toc59612171)

[2. Diagram cho bài toán 32](#_Toc59612172)

[3. Mô hình Convolution Neural Network cụ thể áp dụng vào bài toán 33](#_Toc59612173)

[4. Xây dựng các layer 34](#_Toc59612174)

[5. Chuẩn bị dữ liệu 37](#_Toc59612175)

[6. Quá trình compile và xây dựng model 38](#_Toc59612176)

[7. Kết quả 40](#_Toc59612176)

[Chương V: Tổng kết 42](#_Toc59612177)

**DANH MỤC HÌNH**

[Hình 1: Machine Learning 10](#_Toc59189084)

[Hình 2: Minh họa phần trăm về nghề nghiệp 11](#_Toc59189085)

[Hình 3: Công cụ tìm kiếm bằng hình ảnh 12](#_Toc59189086)

[Hình 4: Phân loại Machine Learning 15](#_Toc59189087)

[Hình 5: Natural Language Processing 17](#_Toc59189088)

[Hình 6: Người máy chơi cờ 18](#_Toc59189089)

[Hình 7: Convolutional Neural Network 19](#_Toc59189090)

[Hình 8: Mạng Neural Network cơ bản 20](#_Toc59189091)

[Hình 9: Sơ đồ hoạt động của CNN 22](#_Toc59189103)

[Hình 10: Sơ đồ hoạt động của CNN với ảnh thực tế 23](#_Toc59189104)

[Hình 11: Ma trận đầu vào (trái) và kernel (phải) 24](#_Toc59189094)

[Hình 12: Kernel quét qua từng phần tử của mảng input 25](#_Toc59189095)

[Hình 13: Kết quả 25](#_Toc59189096)

[Hình 14: Ví dụ thực tế, trước và sau 26](#_Toc59189097)

[Hình 15: Mô hình tổng quát của Conv layer 27](#_Toc59189098)

[Hình 16: Minh họa toán học 27](#_Toc59189099)

[Hình 17: Minh họa Max Pooling (1) 28](#_Toc59189100)

[Hình 18: Minh họa Max pooling (2) 28](#_Toc59189101)

[Hình 19: Sơ đồ Fully – Connected 29](#_Toc59189102)

[Hình 20: Đồ thị hàm Sigmoid 30](#_Toc59189092)

[Hình 21: Đồ thị hàm ReLU 31](#_Toc59189093)

[Hình 22: Planning 32](#_Toc59189106)

[Hình 23: Diagram 32](#_Toc59189105)

[Hình 24: Mô hình CNN áp dụng vào bài toán 33](#_Toc59189107)

[Hình 25: Layer 1&2 34](#_Toc59189108)

[Hình 26: Layer 3 35](#_Toc59189109)

[Hình 27: Layer 4 35](#_Toc59189110)

[Hình 28: Layer 5 35](#_Toc59189111)

[Hình 29: Layer 6-10 35](#_Toc59189112)

[Hình 30: Layer 11 36](#_Toc59189113)

[Hình 31: Layer 12 36](#_Toc59189114)

[Hình 32: Layer 13-15 36](#_Toc59189115)

[Hình 33: Compile & Build Model 38](#_Toc59189116)

[Hình 34: Kết quả sau khi train 38](#_Toc59189117)

Hình 35: Kết quả cuối cùng sau khi test 39

Hình 36: Kết quả url 40

Hình 37: Kết quả json 41

Hình 38: Kết quả giao diện 41

**DANH MỤC BẢNG**

[Bảng 1: Thống kê model 40](#_Toc532425069)

**Lời mở đầu**

Công nghệ Thông tin ngày nay không ngừng phát triển mạnh mẽ. Sự ra đời của công nghệ thông tin đã góp phần giúp cho xã hội và đời sống của con người vươn lên một tầm cao mới, rõ rệt, ảnh hưởng đặc biệt đến sự phát triển của nhân loại.

Những năm gần đây, AI - Artificial Intelligence (Trí Tuệ Nhân Tạo), và cụ thể hơn là Machine Learning (Học Máy hoặc Máy Học) nổi lên như một bằng chứng của cuộc cách mạng công nghiệp lần thứ tư (1 - động cơ hơi nước, 2 - năng lượng điện, 3 - công nghệ thông tin). Trí Tuệ Nhân Tạo đang len lỏi vào mọi lĩnh vực trong đời sống mà có thể chúng ta không nhận ra. Xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trong ảnh của Facebook, trợ lý ảo Siri của Apple, hệ thống gợi ý sản phẩm của Amazon, hệ thống gợi ý phim của Netflix, máy chơi cờ vây AlphaGo của Google DeepMind, …, chỉ là một vài trong vô vàn những ứng dụng của AI/Machine Learning.

Machine Learning là một tập con của AI. Theo định nghĩa của Wikipedia, Machine learning is the subfield of computer science that “gives computers the ability to learn without being explicitly programmed”. Nói đơn giản, 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.



Hình 1: Machine Learning

# Chương I: Tổng quan bài toán phân loại nghề nghiệp

* Đồ án: Tìm hiểu học sâu cho bài toán phân loại nghề nghiệp.
* Mô tả:
* Đề tài: Nhận diện hình ảnh .jpg, sau đó trả ra kết quả phần trăm nghề nghiệp.
* Kiến thức: áp dụng những kiến thức cơ bản của Neural Networks, kiến thức của một số thuật toán, activation function, cost functions để phục vụ cho bài toán phân loại nghề nghiệp.
* Kết quả: sản phẩm là chương trình được viết bằng ngôn ngữ lập trình C#, giao diện Winform, kết hợp API (Application Programming Interface) được viết bằng ngôn ngữ lập trình Python. Phần training model được viết bằng ngôn ngữ Python, sử dụng bộ image data set để đào tạo và đánh giá.



Hình 2: Minh họa phần trăm về nghề nghiệp

1. Lý do lựa chọn đề tài

* Hoàn cảnh:
* **Google:** Cụm từ “Google” có thể rất quen thuộc với tất cả chúng ta, đặc biệt là những ai đã và đang học Công nghệ thông tin. Google được dựa trên một công trình nghiên cứu vào năm 1996 và phát triển cho đến bây giờ. Đến năm 2001, Google phát triển thêm công cụ tìm kiếm mới bằng hình ảnh, cho phép người dùng tìm hình ảnh trên các trang web. Từ đó, em thấy rằng hình ảnh là nguồn dữ liệu đầu vào đa dạng, vì nhiều có nhiều loại và nội dung hình ảnh khác nhau. Trong số đó là về nghề nghiệp, em đặt câu hỏi “vậy hình ảnh nhiều vô kể, làm sao để phân loại được nghề nghiệp ? ”.
* **Công nghệ:** Deep Machine Learning nói riêng và AI (Artificial Intelligence) nói chung trong tương lai đang 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,…...



Hình 3: Công cụ tìm kiếm bằng hình ảnh

* Lý do tại sao chọn vấn đề về phân loại nghề nghiệp qua hình ảnh:
* Vấn đề tìm kiếm thông tin rất phổ biến ở nước ta khi mà tình trạng định hướng nghề nghiệp và hướng nghiệp của các bạn học sinh, sinh viên ngày càng tăng cao.
* Đây là một vấn đề có ở hầu hết các bạn trẻ và tốn nhiều thời gian để xác định, định hướng và độ hiệu quả được nâng cao khi được áp dụng công nghệ AI.
* Nếu hệ thống được cộng đồng chú ý và có độ chính xác cao, chúng ta có thể áp dụng nó vào những lĩnh vực khác trong các ngành: Thương mại điện tử, Công nghiệp trò chơi, Công nghiệp ô tô, Sản xuất, Giáo dục......

## Ý nghĩa đề tài

* Áp dụng Machine Learning & AI vào trong đời sống sẽ giúp rút ngắn thời gian khám phá và định hướng của các bạn trẻ. Điều này sẽ tiết kiệm công sức, thời gian và chi phí rất nhiều.
* Với bài toán phân loại nghề nghiệp, các bạn trẻ sẽ nhanh chóng có được từ khóa tìm kiếm và nội dung nghề nghiệp trên mạng
* Hỗ trợ các học sinh, sinh viên trong việc lựa chọn ngành nghề phù hợp.

## Kết quả mong đợi

* Đáp ứng được yêu cầu đề ra của giáo viên và người hướng dẫn.
* Học hỏi thêm những kiến thức và công nghệ mới.
* Kết quả đầu ra có độ chính xác cao (>80%).
* Áp dụng vào các hệ thống lớn trong tương lai.

# Chương II: Tổng quan về học sâu

1. Tổng quan học máy

Học máy (Machine Learning) là một ngành khoa học nghiên cứu các thuật toán cho phép máy tính có thể học được các khái niệm (concept).

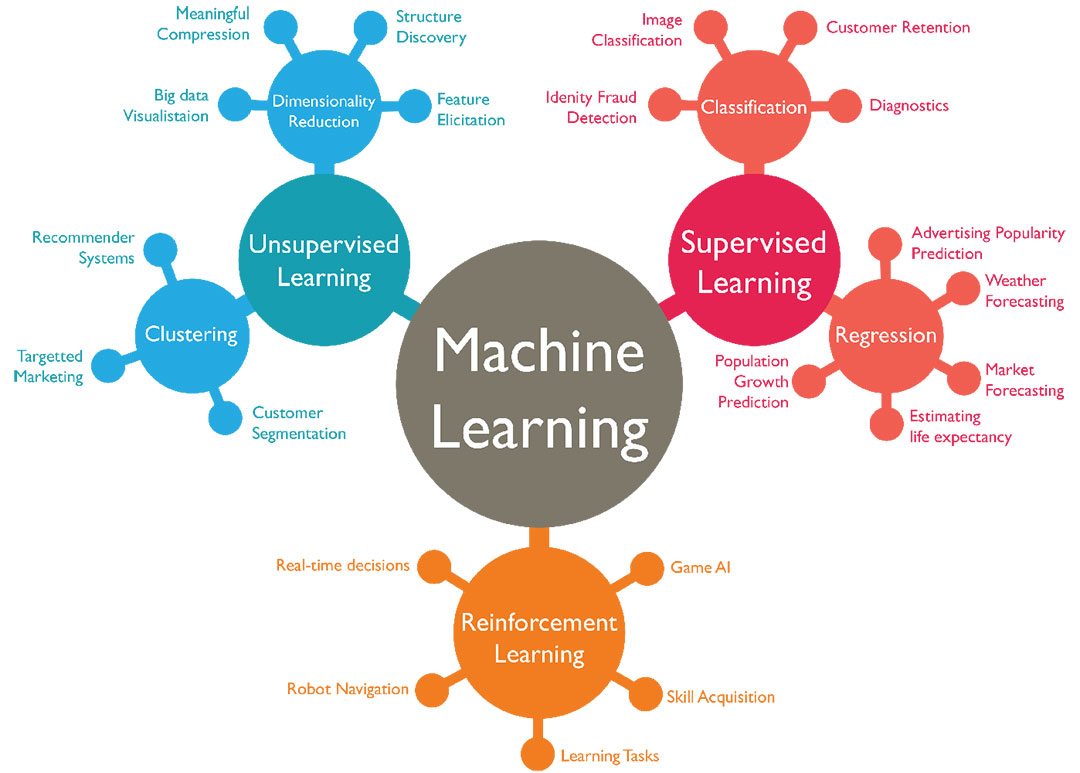
Phân loại: Có hai loại phương pháp học máy chính

* **Phương pháp quy nạp:** Máy học/phân biệt các khái niệm dựa trên dữ liệu đã thu thập được trước đó. Phương pháp này cho phép tận dụng được nguồn dữ liệu rất nhiều và sẵn có.
* **Phương pháp suy diễn:** Máy học/phân biệt các khái niệm dựa vào các luật. Phương pháp này cho phép tận dụng được các kiến thức chuyên ngành để hỗ trợ máy tính.

Hiện nay, các thuật toán đều cố gắng tận dụng được ưu điểm của hai phương pháp này.

Các ngành khoa học liên quan:

* Lý thuyết thống kê: các kết quả trong xác suất thống kê là tiền đề cho rất nhiều phương pháp học máy. Đặc biệt, lý thuyết thống kê cho phép ước lượng sai số của các phương pháp học máy.
* Các phương pháp tính: các thuật toán học máy thường sử dụng các tính toán số thực/số nguyên trên dữ liệu rất lớn. Trong đó, các bài toán như: tối ưu có/không ràng buộc, giải phương trình tuyến tính, v.v… được sử dụng rất phổ biến.
* Khoa học máy tính: là cơ sở để thiết kế các thuật toán, đồng thời đánh giá thời gian chạy, bộ nhớ của các thuật toán học máy.



Hình 4: Phân loại Machine Learning

Các nhóm giải thuật học máy:

* **Học có giám sát (Supervised Learning):** Máy tính được xem một số mẫu gồm đầu vào (input) và đầu ra (output) tương ứng trước. Sau khi học xong các mẫu này, máy tính quan sát một đầu vào mới và cho ra kết quả.
* **Học không giám sát (Unsupervised Learning):** Máy tính chỉ được xem các mẫu không có đầu ra, sau đó máy tính phải tự tìm cách phân loại các mẫu này và các mẫu mới.
* **Học nửa giám sát (Semi-Supervised Learning):** Một dạng lai giữa hai nhóm giải thuật trên.
* **Học tăng cường (Reinforcement Learning):** Máy tính đưa ra quyết định hành động (action) và nhận kết quả phản hồi (response/reward) từ môi trường (environment). Sau đó máy tính tìm cách chỉnh sửa cách ra quyết định hành động của mình.

1. Tổng quan học sâu

Học sâu là một phần của một họ các phương pháp [học máy](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y) rộng hơn dựa trên [đại diện học](https://vi.wikipedia.org/w/index.php?title=Learning_representation&action=edit&redlink=1) của dữ liệu. Một quan sát (ví dụ như, một hình ảnh) có thể được biểu diễn bằng nhiều cách như một [vector](https://vi.wikipedia.org/wiki/Kh%C3%B4ng_gian_vect%C6%A1) của các giá trị cường độ cho mỗi điểm ảnh, hoặc một cách trừu tượng hơn như là một tập hợp các cạnh, các khu vực hình dạng cụ thể, vv. Một vài đại diện làm khiến việc học các nhiệm vụ dễ dàng hơn (ví dụ, nhận dạng khuôn mặt hoặc biểu hiện cảm xúc trên khuôn mặt) từ các ví dụ. Một trong những hứa hẹn của học sâu là thay thế các tính năng thủ công bằng các thuật toán hiệu quả đối với [học không có giám sát](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_kh%C3%B4ng_c%C3%B3_gi%C3%A1m_s%C3%A1t) hoặc [nửa giám sát](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_n%E1%BB%ADa_gi%C3%A1m_s%C3%A1t) và tính năng phân cấp.

Các nghiên cứu trong lĩnh vực này cố gắng thực hiện các đại diện tốt hơn và tạo ra các mô hình để tìm hiểu các đại diện này từ dữ liệu không dán nhãn quy mô lớn. Một số đại diện được lấy cảm hứng bởi những tiến bộ trong [khoa học thần kinh](https://vi.wikipedia.org/wiki/Khoa_h%E1%BB%8Dc_th%E1%BA%A7n_kinh) và được dựa trên các giải thích của mô hình xử lý và truyền thông thông tin trong một [hệ thống thần kinh](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BA%A7n_kinh), chẳng hạn như [mã hóa thần kinh](https://vi.wikipedia.org/w/index.php?title=M%C3%A3_h%C3%B3a_th%E1%BA%A7n_kinh&action=edit&redlink=1) để cố gắng để xác định các mối quan hệ giữa các kích thích khác nhau và các phản ứng liên quan đến thần kinh trong [não](https://vi.wikipedia.org/wiki/N%C3%A3o).

Nhiều kiến trúc học sâu khác nhau như [mạng neuron sâu](https://vi.wikipedia.org/w/index.php?title=M%E1%BA%A1ng_n%C6%A1-ron_s%C3%A2u&action=edit&redlink=1), [mã mạng neuron tích chập sâu](https://vi.wikipedia.org/w/index.php?title=Convolutional_neuron_network&action=edit&redlink=1), [mạng niềm tin sâu](https://vi.wikipedia.org/w/index.php?title=Deep_belief_network&action=edit&redlink=1) và [mạng neuron tái phát](https://vi.wikipedia.org/w/index.php?title=Recurrent_neuron_network&action=edit&redlink=1) đã được áp dụng cho các lĩnh vực như [thị giác máy tính](https://vi.wikipedia.org/wiki/Th%E1%BB%8B_gi%C3%A1c_m%C3%A1y_t%C3%ADnh), [tự động nhận dạng giọng nói](https://vi.wikipedia.org/wiki/Nh%E1%BA%ADn_d%E1%BA%A1ng_ti%E1%BA%BFng_n%C3%B3i), [xử lý ngôn ngữ tự nhiên](https://vi.wikipedia.org/wiki/X%E1%BB%AD_l%C3%BD_ng%C3%B4n_ng%E1%BB%AF_t%E1%BB%B1_nhi%C3%AAn), nhận dạng âm thanh ngôn ngữ và [tin sinh học](https://vi.wikipedia.org/wiki/Tin_sinh_h%E1%BB%8Dc), chúng đã được chứng minh là tạo ra các kết quả rất tốt đối với nhiều nhiệm vụ khác nhau.

## Các ứng dụng của học sâu

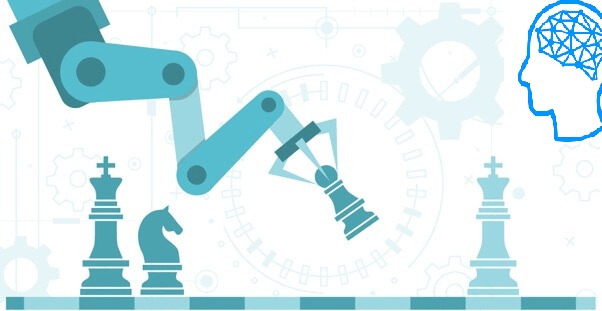
Học sâu có ứng dụng rộng khắp trong các ngành khoa học/sản xuất, đặc biệt những ngành cần phân tích khối lượng dữ liệu khổng lồ. Một số ứng dụng thường thấy:

* Xử lý ngôn ngữ tự nhiên (Natural Language Processing): xử lý văn bản, giao tiếp người – máy,…



Hình 5: Natural Language Processing

* Nhận dạng (Pattern Recognition): nhận dạng tiếng nói, chữ viết tay, vân tay, thị giác máy (Computer Vision).
* Tìm kiếm (Search Engine).
* Chẩn đoán trong y tế: phân tích ảnh X-quang, các hệ chuyên gia chẩn đoán tự động.
* Tin sinh học: phân loại chuỗi gene, quá trình hình thành gene/protein.
* Vật lý: phân tích ảnh thiên văn, tác động giữa các hạt.
* Phát hiện gian lận tài chính (financial fraud): gian lận thẻ tỉn dụng.
* Phân tích thị trường chứng khoán (stock market analysis).
* Chơi trò chơi: tự động chơi cờ, hành động của các nhân vật ảo.



Hình 6: Người máy chơi cờ

Rôbốt: là tổng hợp của rất nhiều ngành khoa học, trong đó học máy tạo nên hệ thần kinh/bộ não của người máy

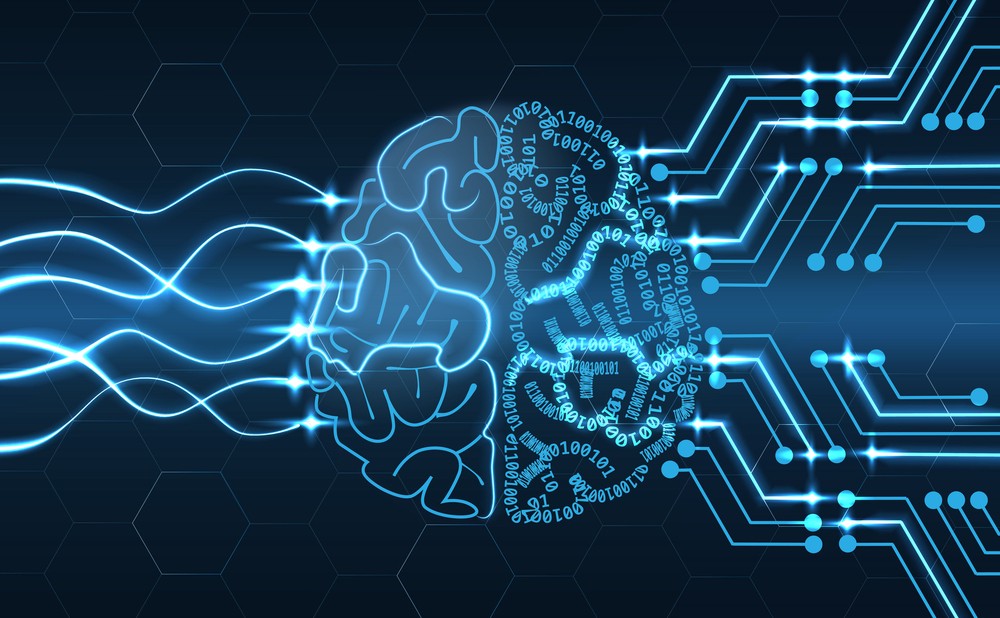
.

# Chương III: Mô hình CNN

Trong bài toán này, nhóm chúng em sẽ sử dụng thuật toán CNN (Convolutional Neural Network) để giải quyết vấn đề. CNN được sử dụng nhiều trong các bài toán nhận dạng các object trong phân ảnh. Để tìm hiểu tại sao thuật toán này được sử dụng rộng rãi cho việc nhận dạng (detection), chúng ta hãy cùng tìm hiểu về thuật toán này.

## Tổng quan về CNN

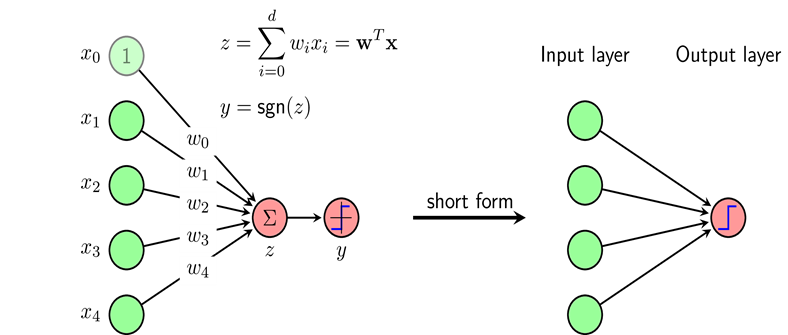
**CNN (Convolutional Neural Network)** là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao như hiện nay như hệ thống xử lý ảnh lớn như Facebook, Google hay Amazon đã đưa vào sản phẩm của mình những chức năng thông minh như nhận diện khuôn mặt người dùng, phát triển xe hơi tự lái hay drone giao hàng tự động. CNN được sử dụng nhiều trong các bài toán nhận dạng các object trong ảnh.Ví dụ như tự động nhận diện khi chúng ta đăng một ảnh lên Facebook, hay khi search một từ bất kì lên google image search, ví dụ "Mèo" thì trong tab "Hình ảnh", google sẽ hiển thị rất nhiều ảnh có mèo trong đó. CNN chính là một dạnh Artificial Neural Network, một Multiplayer Perceptron nhưng mang thêm 1 vài cải tiến, đó là **Convolution** và **Pooling**.



Hình 7: Convolutional Neural Network

1. Đặc điểm của CNN
   1. Áp dụng Neural Network

Mạng nơron nhân tạo, Convolutional Neural Network (CNN) 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.



Hình 8: Mạng Neural Network cơ bản

Các thành phần cơ bản của một nơron nhân tạo bao gồm:

* Tập các đầu vào: Là các tín hiệu vào (input signals) của nơron, các tín hiệu này thường được đưa vào dưới dạng một vector N chiều.
* Tập các liên kết: Mỗi liên kết đƣợc thể hiện bởi một trọng số liên kết – Synaptic weight. Trọng số liên kết giữa tín hiệu vào thứ j với nơron k thường được kí hiệu là w\_kj. Thông thường, các trọng số này được khởi tạo một cách ngẫu nhiên ở thời điểm khởi tạo mạng và được cập nhật liên tục trong quá trình học mạng.
* 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.
* Đầu ra: Là tín hiệu đầu ra của một nơron, với mỗi nơron sẽ có tối đa là một đầu ra.
  1. Ưu và nhược điểm của CNN

Trong thực tế, mỗi bức ảnh đầu vào của một bài toán có kích thước là D \* R \* C (D là chiều dài, R là chiều rộng, C là số màu) thì số lượng tham số cần phải xử lý là rất lớn. Giả sử một bức hình đơn giản sẽ có 28x28x3=2352 tham số, đây là một con số rất lớn nhưng trong thực tế thì kích thước của một bức ảnh còn lớn hơn rất nhiều (Đối với một bức ảnh X-ray thông thường thì chiều dài và chiều rộng ít nhất là 425 và 512). Và CNN sẽ giải quyết được vấn đề này, cái mà ANN không thể làm được. Vì sao?Vì CNN sẽ giúp số tham số đầu vào của ANN trở nên ít lại nhưng vẫn giữ nguyên các thuộc tính cần thiết cho bức hình.

Nhưng CNN gặp phải các khó khăn sau:

* Cần lượng dữ liệu rất lớn để có thể xây dựng một model chuẩn.
* Tiêu tốn nhiều tài nguyên để thực hiện thuật toán.
* Các bức ảnh đầu vào phải có cùng một kích thước (width, height, dimension,…).
* Khi các bức hình đầu vào có kích thước lớn hơn thì kéo theo số lượng layer tăng rất nhiều.

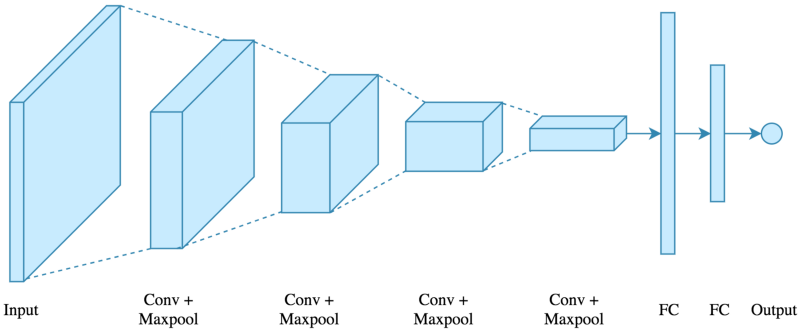
## Cách hoạt động của CNN

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):

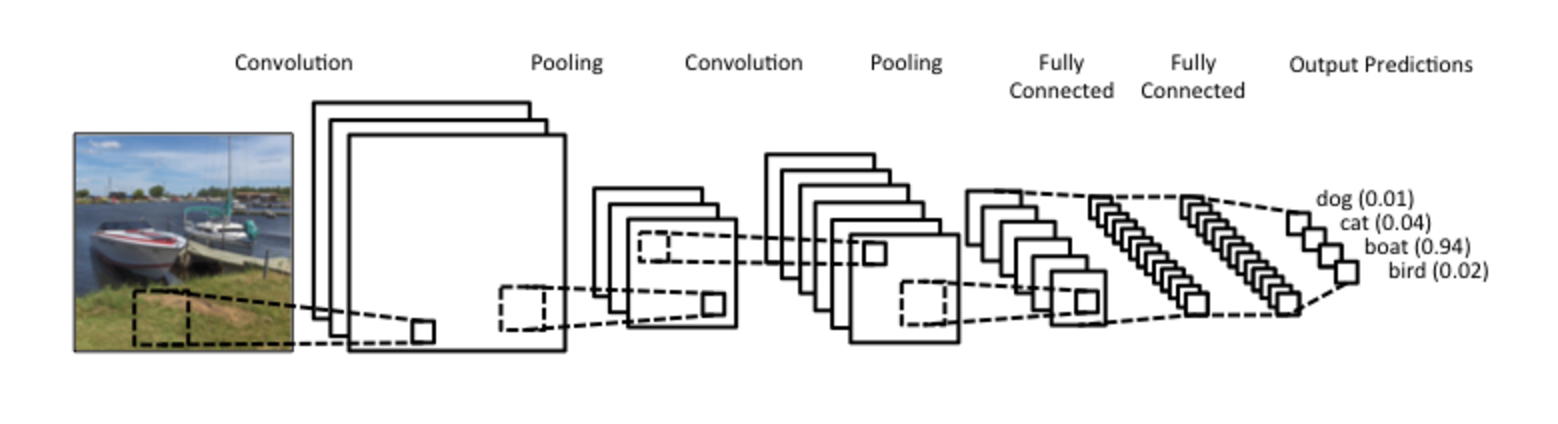
[[CONV 🡪 RELU]\*N 🡪 POOL?]\*M 🡪 [FC 🡪 RELU]\*K 🡪 FC

Trong đó:

* **[CONV -> RELU]\*N** tức là trong kiến trúc này sau tầng CONV là tầng RELU, trong CNN kiến trúc 2 tầng này có thể lặp N lần.
* **POOL?** là tầng Pooling cho người thiết kế quyết định có thể có hoặc không.
* **[[CONV -> RELU]\*N -> POOL?]\*M** trong kiến trúc CNN có thể lặp lại M lần kiểu sau tầng CONV là tầng RELU và kế tới là tầng Pooling.
* **[FC -> RELU]\*K** trong CNN có thể lặp K lần cấu trúc kiểu sau tầng FC là tầng RELU nhưng trước nó phải có tầng [CONV -> RELU].



Hình 9: Sơ đồ hoạt động của CNN



Hình 10: Sơ đồ hoạt động của CNN với ảnh thực tế

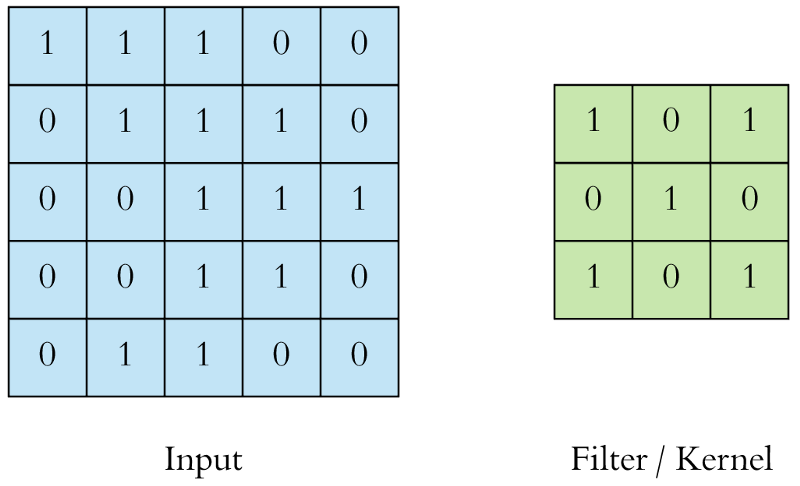
Trong suốt quá trình huấn luyện, CNNs sẽ tự động học được các thông số.

## Các thành phần cơ bản của mạng CNN

1. Convolution

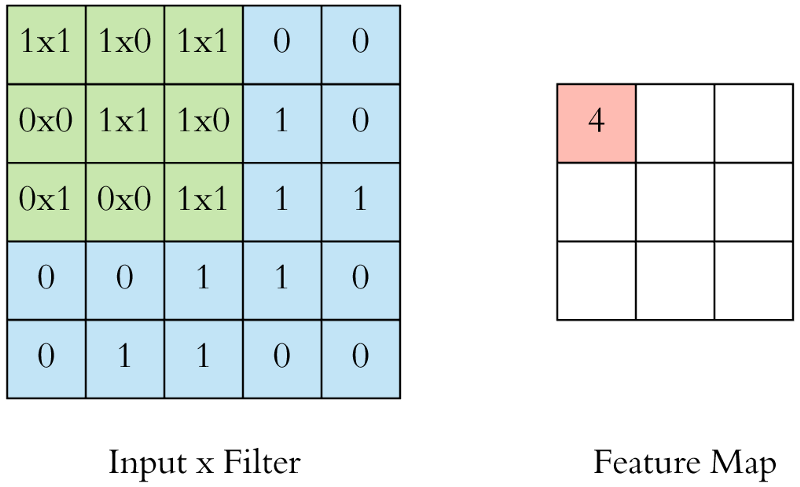
Convolution gồm 2 khái niệm khác là **Convolution Filter** và **Convolutional Layer.**Trong mạng Neural Network thông thường, từ input, ta cho qua các Hidden Layer rồi ra được output. Với CNN, Convolutional Layer cũng chính là Hidden Layer, khác ở chỗ, Convolutional Layer là một tập các feature map và mỗi feature map này là một bản scan của input ban đầu, nhưng được trích xuất ra các feature/đặc tính cụ thể. Scan như thế nào thì lại dựa vào Convolution Filter hay kernel. Đây là một ma trận sẽ quét qua ma trận dữ liệu đầu vào, từ trái qua phải, trên xuống dưới, và nhân tương ứng từng giá trị của ma trận đầu vào mà ma trận kernel rồi cộng tổng lại, đưa qua activation funciton (sigmoid, relu, elu, ... ), kết quả sẽ là một con số cụ thể, tập hợp các con số này lại là 1 ma trận nữa, chính là feature map.

Ví dụ: có một ma trận đầu vào input và một kernel.



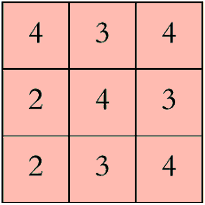
Hình 11: Ma trận đầu vào (trái) và kernel (phải)

Quét kernel qua từng phần tử của input. Và tính toán như trên: nhân tương ứng, rồi cộng tổng kết quả, đưa qua activation function, ta thu được một giá trị tại feature map:



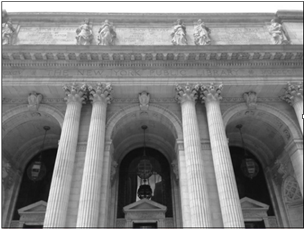
Hình 12: Kernel quét qua từng phần tử của mảng input

Thực hiện lần lượt cho đến hết, ta thu được Feature Map cuối cùng:



Hình 13: Kết quả

Chúng ta không cần phải quan tâm tới việc phải tìm bao nhiêu kernel hay lưu các kernel về để dùng dần. Đó là việc của CNN, nó sẽ tự động tìm các kernel, tự dò ra các feature.



Hình 14: Ví dụ thực tế, trước và sau

* **Chi tiết về Convolution Layer:** Là một layer đầu tiên để tách features từ một bức hình đầu vào. Tích chập bảo toàn mối quan hệ giữa các điểm ảnh bằng cách học features của bức hình sử dụng những hình vuông nhỏ của dữ liệu đầu vào. Nó là một thuật toán mà xem 2 dữ liệu đầu vào là ma trận hình ảnh và một kernel.

Một số khái niệm cơ bản: **Filter**, **Kernel** hay **FeatureDetector** đều là cách gọi của ma trận lọc (như đã đề cập ở trên). Thông thường, ở các lớp đầu tiên của Conv Layer sẽ có kích thước là [5x5x3]:

- Convolved Feature, Activation Map hay Feature Map là đầu ra của ảnh khi cho bộ lọc chạy hết bức ảnh với phép tích vô hướng.

- Receptive field là vùng ảnh được chọn để tính tích chập, hay bằng đúng cái kích thước của bộ lọc.

- Depth là số lượng bộ lọc.

- Stride được hiểu là khoảng cách dịch chuyển của bộ lọc sau mỗi lần tính. Ví dụ khi stride=2. Tức sau khi tính xong tại 1 vùng ảnh, nó sẽ dịch sang phải 2 pixel. Tương tự với việc dịch xuống dưới.

- Zero-Padding là việc thêm các giá trị 0 ở xung quanh biên ảnh, để đảm bảo phép tích chập được thực hiện đủ trên toàn ảnh.

Cách tính kích thước đầu ra của ảnh mỗi layer:

Giả sử ảnh đầu ra là [ x x ]

Thì:

=

=

=

Trong đó:

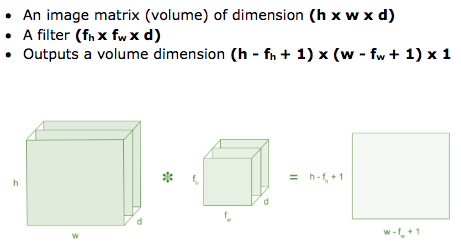
[ x x ]: Kích thước đầu vào

: Kích thước bộ lọc Kernel

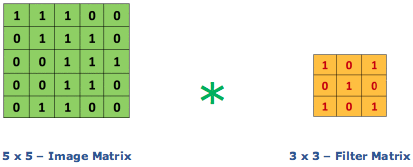
: giá trị Stride

: số lượng zero-padding thêm vào viền ảnh

: Số lượng bộ lọc (Depth)



Hình 15: Mô hình tổng quát của Conv layer



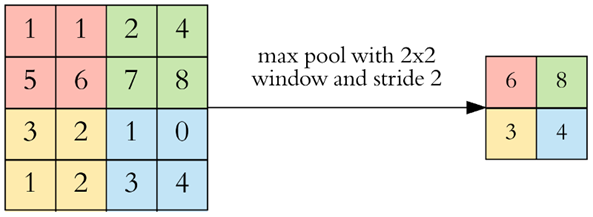
Hình 16: Minh họa toán học

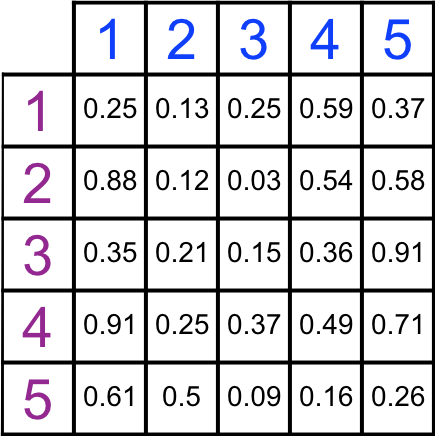
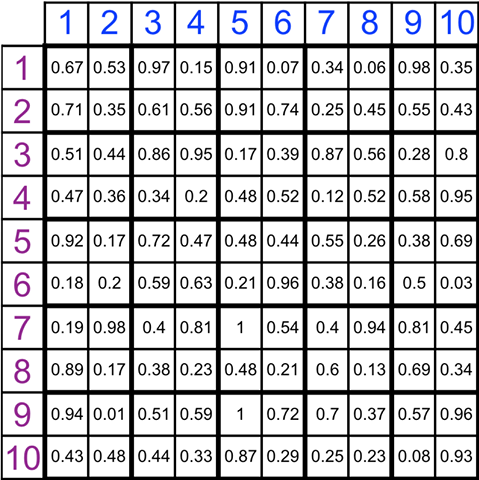
1. Max pooling

Pooling được sử dụng trong CNN để giảm số hyperparameter cần phải tính toán, từ đó giảm thời gian tính toán, tránh overfitting. Có 2 loại Pooling: Max pooling (dùng nhiều nhất) và Average pooling.

Pooling layer cung cấp tính bất biến đối với phép dịch chuyển (translation), phép quay (rotation) và phép co giãn (scaling).

Trong bài toán này, chúng ta sẽ sử dụng Max pooling: giữ lại chi tiết quan trọng hay hiểu ở trong bài toán này chính giữ lại pixel có giá trị lớn nhất.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.

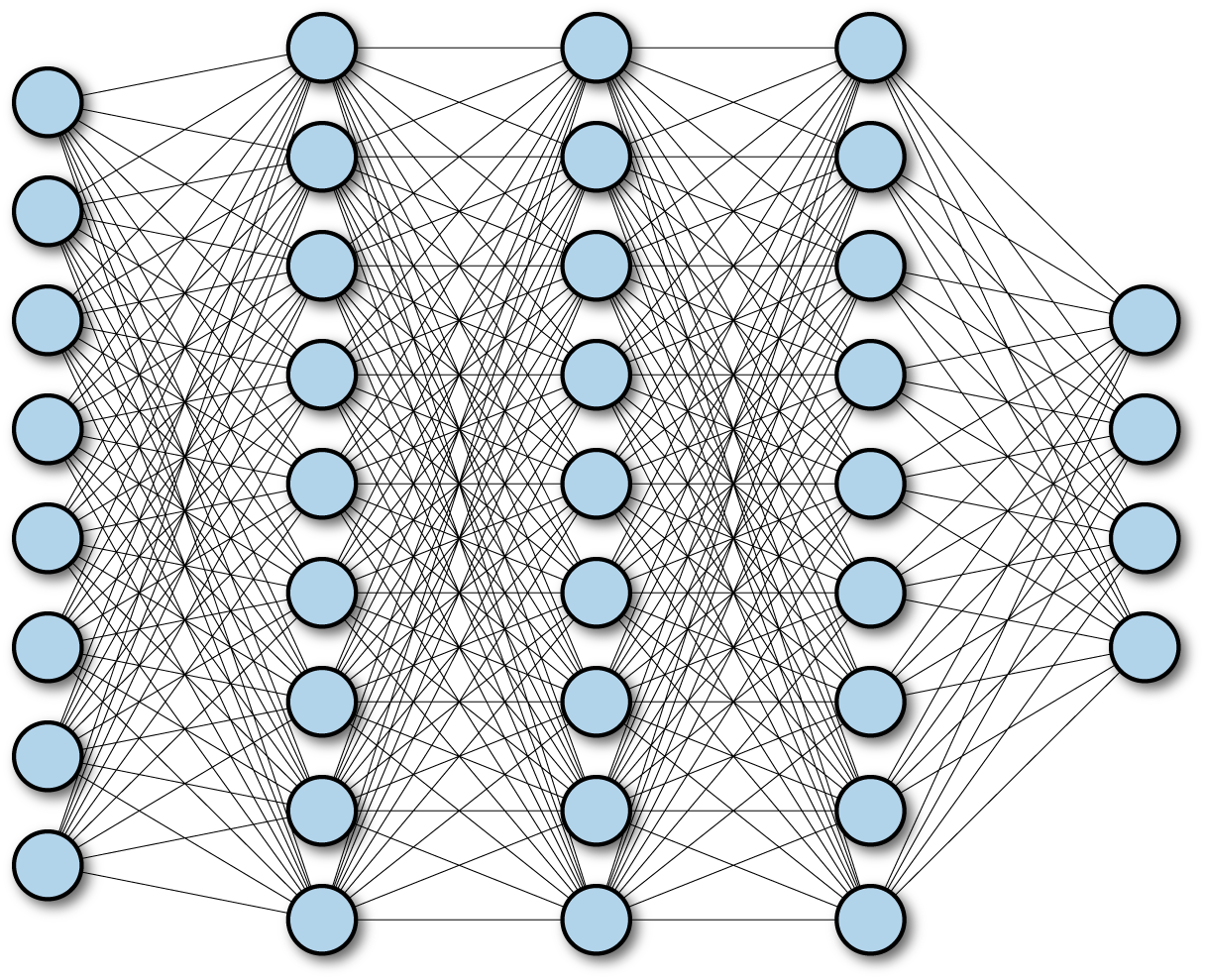
****

Hình 17: Minh họa Max Pooling (1)

Hình 18: Minh họa Max pooling (2)

1. Fully – connected

Mục đích: kết nối mọi neuron trong một lớp với mọi neuron trong một lớp khác. Về nguyên tắc, nó giống như multi-layer perceptron neural network truyền thống.Thực hiên sau convolutional layers + pooling layers. Vì ouput của 2 layer này có kích thước 3D, còn output của fully connected layer là mảng 1D nên phải Flatten output của pooling layer cuối cùng. Flatten chỉ đơn giản là sắp xếp khối 3D của các số thành một vector 1D ( n hàng, 1 cột).



Hình 19: Sơ đồ Fully– Connected

## Hàm kích hoạt (activation function)

Activation function giới hạn phạm vi giá trị tới một giá trị hữu hạn cho phép.Một số hàm activation thường dùng : Sigmoid, Rectified linear unit (ReLU), Softmax.

1. Sigmoid function

Sigmoid function là một logistic function có phạm vi (0, 1). Do đó, nó đặc biệt được sử dụng cho các mô hình mà phải dự đoán xác suất. Vì vậy xác suất của bất cứ điều gì chỉ tồn tại giữa phạm vi 0 và 1.

Áp dụng tốt nhất có bài toán classification với chỉ hai class.



Hình 20: Đồ thị hàm Sigmoid

1. Softmax

Hàm softmax có output nằm trong khoảng từ 0 đến 1, giống như hàm Sigmoid. Nhưng nó chia mỗi output sao cho tổng số output bằng 1.

1. Rectified linear unit (ReLU)

ReLU (Rectified Linear Unit) được sử dụng rộng rãi gần đây vì tính đơn giản của nó. Nó có công thức toán học - rất đơn giản. Ưu điểm chính của nó là:

* ReLU được chứng minh giúp cho việc training các Deep Networks nhanh hơn rất nhiều. Sự tăng tốc này được cho là vì ReLU được tính toán gần như tức thời và gradient của nó cũng được tính cực nhanh với gradient bằng 1 nếu đầu vào lớn hơn 0, bằng 0 nếu đầu vào nhỏ hơn 0.
* Mặc dù hàm ReLU không có đạo hàm tại , trong thực nghiệm, người ta vẫn thường định nghĩa ReLU và khẳng định thêm rằng, xác suất để input của một unit bằng 0 là rất nhỏ.

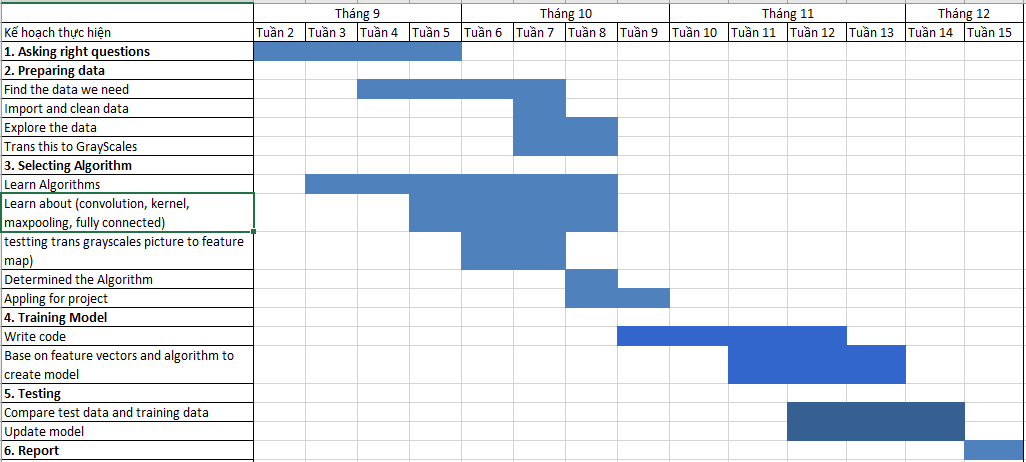


Hình 21: Đồ thị hàm ReLU

Với bất kỳ input có giá trị âm đưa vào cho hàm ReLU sẽ biến giá trị thành 0, do đó ảnh hưởng đến biểu đồ kết quả khi không ánh xạ các giá trị âm một cách thích hợp.

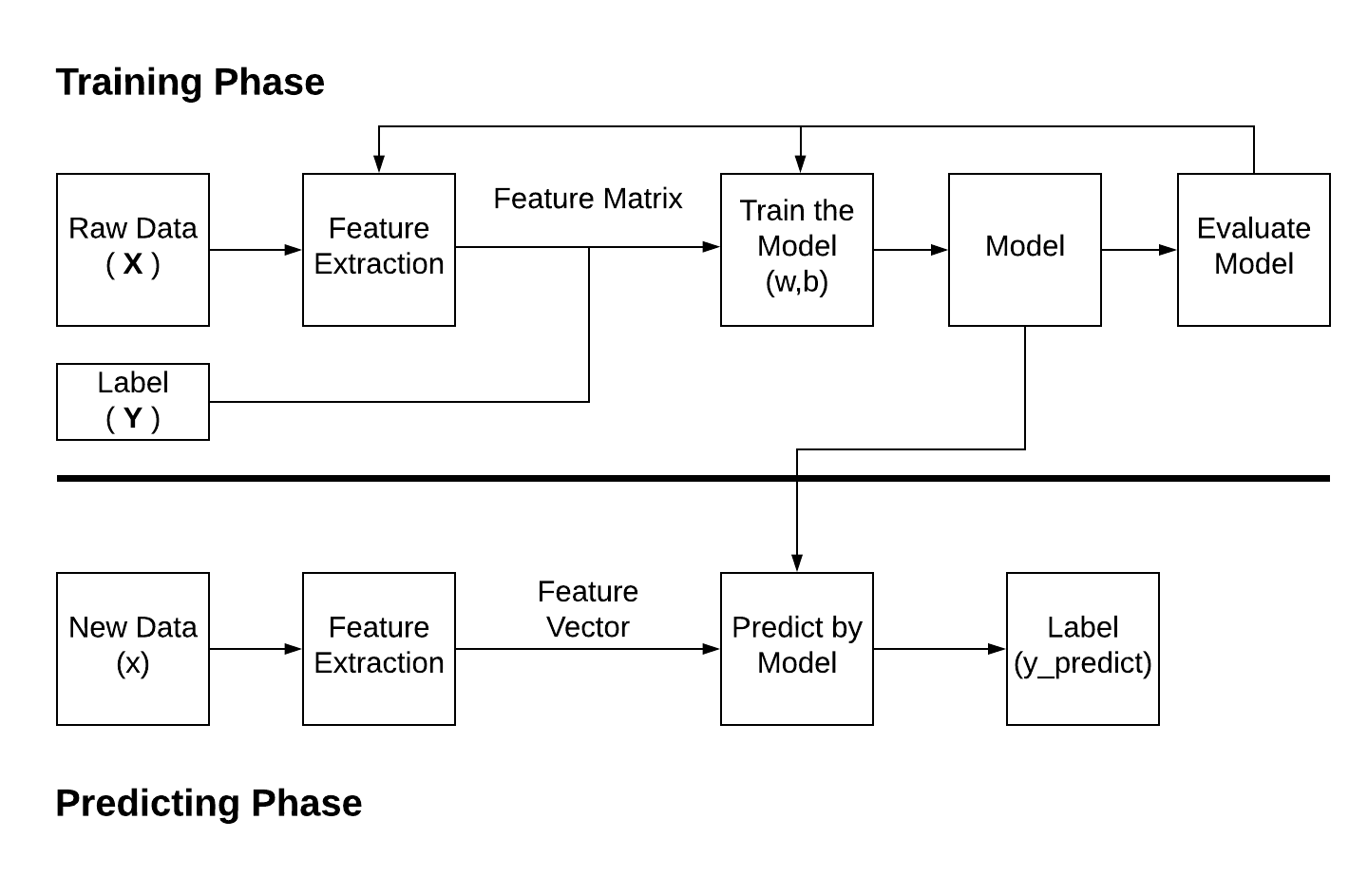
# Chương IV: Áp dụng học sâu vào bài toán

## 1. Kế hoạch



Hình 22: Planning

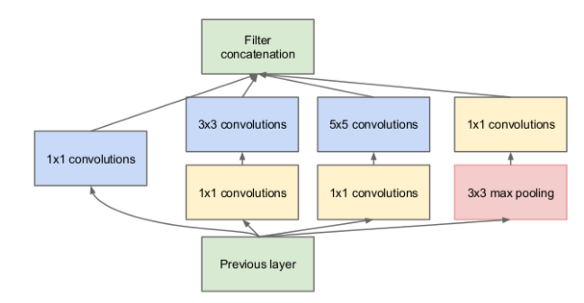
## Diagram cho bài toán



Hình 23: Diagram

## Mô hình Convolution Neural Network cụ thể áp dụng vào bài toán

Số neuron của Previous Layer bằng số class phân loại.Kết quả nhắm đến của bài toán này là xác định các mảnh hình ảnh.



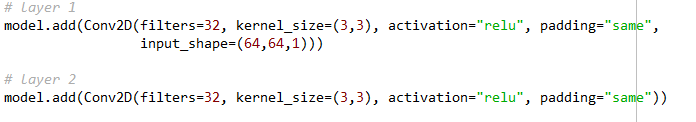
Hình 24: Mô hình CNN áp dụng vào bài toán

## Xây dựng các layer

Thông thường, một CNN gồm một stack các convolutional module, mỗi một module thực hiện việc extract feature. Mỗi module cũng bao gồm một convolutional layer theo sau là một pooling layer. Module convolutional cuối cùng bao gồm một hoặc nhiều các dense layer thực hiện việc phân lớp.Layer dense cuối cùng trong một CNN bao gồm một single node cho mỗi một lớp cụ thể trong model (tất cả các class khả thi mà model có thể dự đoán được). Với mỗi một node, ta sử dụng hàm kích hoạt softmax để generate ra giá trị predict (từ 0 - 1).

Trong bài toán này, nhóm sử dụng tổng cộng là 15 layer.

* Layer 1 & 2:
* Dùng convolution layer (*Conv2D*) dùng để lấy feature từ image, trong đó:
* *filters:* số filter của convolution layer
* *kernel\_size:* size Sliding window trượt trên image
* *relu:* max(0,x) dùng trong các layer cnn để giảm chi phí tính toán. Có tác dụng đưa các giá trị âm về thành 0. Để loại bỏ các giá trị âm không cần thiết mà có thể sẽ ảnh hưởng cho việc tính toán ở các layer sau đó.
* *padding="same":* có sử dụng padding (="valid": không dùng)
* *input\_shape:* chính là kích thước của dữ liệu đầu vào.
* Layer 1 đầu tiên là layer input nên có *input\_shape.*



Hình 25: Layer 1&2

* Layer 3: *BatchNormalization()* giúp giảm lượng thông số cần tính toán lại trên mỗi một feature. Đối với mỗi feature thông thường, giá trị sẽ chạy từ 1-255, nhưng khi có hàm BatchNormalization giá trị của mỗi feature chỉ chạy từ 0-1 (hệ thập phân).



Hình 26: Layer 3

* Layer 4:
* *MaxPooling2D:* lấy những đặc điểm nổi bật nhất và resize lại ảnh.
* *pool\_size:* size pooling, thường có size ma trận 2x2, đối với ảnh lớn thì 4x4.



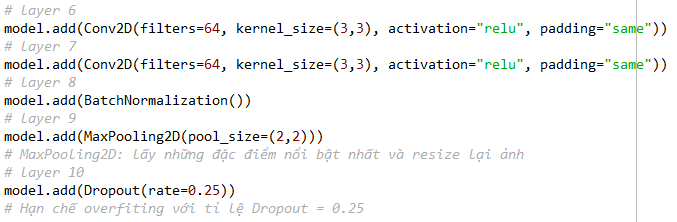
Hình 27: Layer 4

* Layer 5: Hạn chế overfiting với tỉ lệ *Dropout = 0.25*.



Hình 28: Layer 5

* Layer 6-10: Từ layer 6-10 chúng ta thực hiện lại quá trình giống từ 1-5, mục đích nhằm tăng số lượng data input qua đó tăng số lượng phép tính và giảm độ phức tạp trên mỗi phép tính.



Hình 29: Layer 6-10

* Layer 11: *Flatten()* dùng để lát phẳng layer để Fully Connection và chuyển thành ma trận cột.



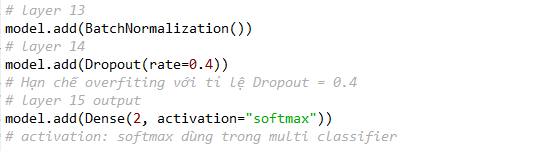
Hình 30: Layer 11

* Layer 12:
* *Dense* layer này sử dụng như một layer neural network bình thường, 1024: chiều output (với filters ảnh đầu vào là 32x32).
* Activation: dùng để chọn *activation="relu"*, relu max(0,x) dùng trong các layer CNN để giảm chi phí tính toán.



Hình 31: Layer 12

* Layer 13-15:*activation: softmax* dùng trong Multi classifier.



Hình 32: Layer 13-15

## Chuẩn bị dữ liệu

Về cơn bản, chúng ta sẽ chia tập dữ liệu thành 3 phần:

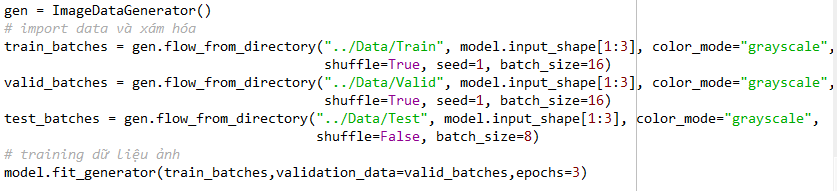
* Train: đây là tập dữ liệu dùng để training. Trong này sẽ chia được chia tiếp thành 10 tập là các ngành nghề.
* Valid: đây là tập dữ liệu dùng để xác thực. Trong này sẽ chia được chia tiếp thành 10 tập là các ngành nghề.
* Test: đây là tập dữ liệu dùng để test (bước cuối). Trong này sẽ chia được chia tiếp thành 10 tập là các ngành nghề.

Dataset được nhóm lấy từ website này:

https://github.com/OlafenwaMoses/IdenProf/releases/download/v1.0/idenprof-jpg.zip

Dataset chia thành 9000 (900 hình cho mỗi nghề) hình ảnh để đào tạo mô hình trí tuệ nhân tạo và 2000 (200 hình cho mỗi nghề) hình ảnh để kiểm tra hiệu suất của mô hình trí tuệ nhân tạo khi nó đang đào tạo.

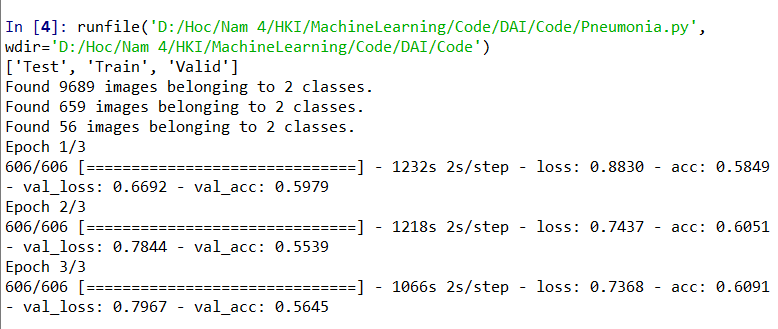
## Quá trình compile và huấn luyện model



Hình 33: Compile & Build Model

Trong mô hình Keras với API chức năng, cần gọi *fit\_generator()* để huấn luyện dữ liệu hình ảnh được tăng cường bằng cách sử dụng *ImageDataGenerator()*. Sau đó, chúng ta nhúng hình ảnh theo từng tập định sẵn là train, valid và test. Đồng thời ta cũng thực hiện tác vụ xám hóa. Ở đây, ta quy định *batch\_size=16*, với mục đích là sẽ có 16 bức hình được chương trình xử lý mỗi lần.

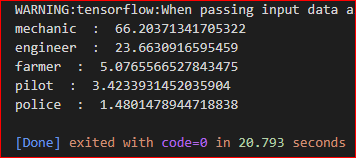
Chương trình sẽ thực thi 3 epoch lần lượt, mỗi epoch gồm 606 tiến trình. Lý do vì trong tập dữ liệu train có tổng cộng là 9689 bức hình và mỗi lần chương trình xử lý 16 bức. Quá trình chạy diễn ra như hình dưới:

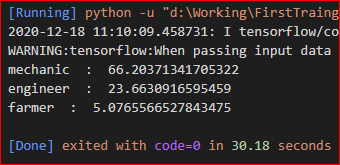


Hình 34: Kết quả sau khi train

Sau khi có được model, chúng ta sẽ chạy file **FirstCustomImageRecognition.py** với model vừa được tạo. Nghĩa là chúng ta sẽ sử dụng số lượng hình ảnh trong tập dữ liệu Test (ở trong ví dụ này là 66) để đánh giá độ chính xác của model.

Kết quả thu được





Hình 35: Kết quả cuối cùng sau khi test

1. ***Kết quả***
   1. **Thống kê model**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Learning rate | Số hình ảnh train | Số hình ảnh test | Tỷ lệ hình ảnh | Độ chính xác |
| MODEL\_1 | 0,001 | 9760 | 29 | 336:1 | 77.6% |
| MODEL\_2 | 0,002 | 9689 | 56 | 173:1 | 78.5% |
| MODEL\_3 | 0,002 | 9689 | 28 | 346:1 | 67.9% |
| MODEL\_4 | 0,001 | 9707 | 38 | 255:1 | 60.5% |

Bảng 1: Thống kê model

* 1. **Xây dựng API**
     + Kết quả URL thu được

API sẽ load model train được và chạy với parameter là hình ảnh .jpg

2020-12-22 19:48:11.492904: I tensorflow/core/platform/cpu\_feature\_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX2

 \* Serving Flask app "app" (lazy loading)

 \* Environment: production

   WARNING: This is a development server. Do not use it in a production deployment.

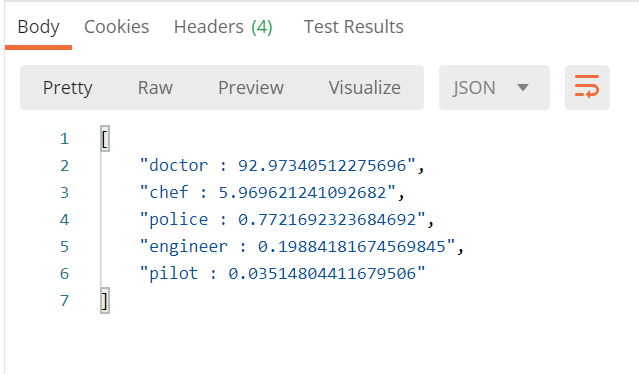
   Use a production WSGI server instead.

 \* Debug mode: off

 \* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

Hình 36: Kết quả url

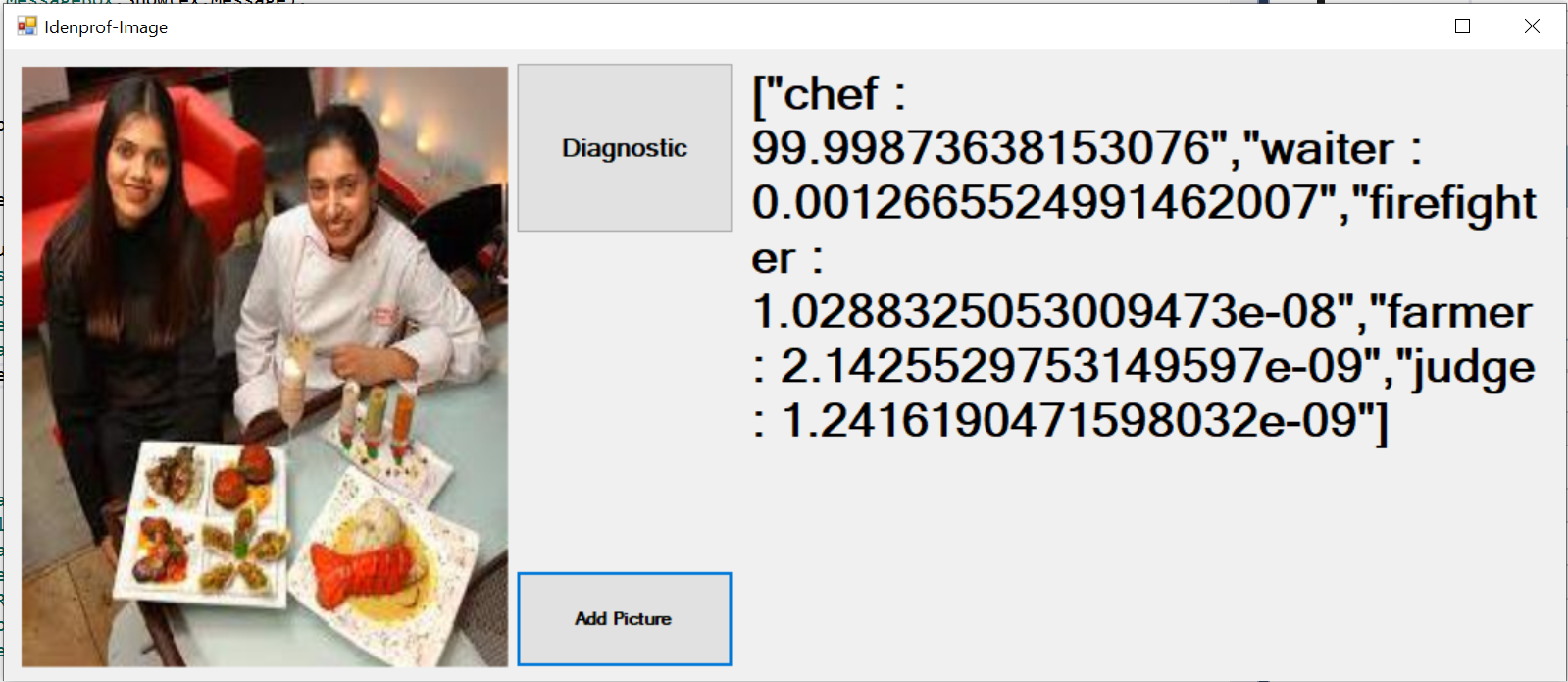
* + - Kết quả JSON thu dược



Hình 37: Kết quả json

* 1. **Xây dựng giao diện Winform**

Dữ liệu nhận được JSON, hiện lên giao diện sau khi ‘Add Image’



Hình38: Kết quả giao diện

# Chương V: Tổng kết

Qua đồ án trên, nhóm đã tiếp cận với các khía cạnh của Deep Machine Learning (Học sâu), hiểu được tầm quan trọng của nó. Biết được Các tiến trình bên trong một thuật toán CNN cơ bản và làm thế nào để sử dụng chúng.Ngoài ra, nhóm cũng hiểu được tiến trình xử lý hình ảnh của máy tính.Dù cho kết quả cuối cùng chưa thực sự tốt nhưng trong tương lai nhóm sẽ cố gắng hoàn thiện sản phẩm hơn.Tổng kết lại, ta có thể thấy trong suốt vài thập niên qua, con người luôn cố gắng vay mượn sức mạnh tính toán của máy tính (một trong những giới hạn của con người) để cải thiện chất lượng cuộc sống cũng như công việc của mình. Machine learning là một trong những công cụ giúp cho chúng ta có thể giải quyết những bài toán mà phương pháp lập trình truyền thống chỉ dựa trên tiến trình không thể nào đạt được. Bằng cách vay mượn một lượng lớn dữ liệu có sẵn, ta có thể “dạy” cho máy tính “học” để chúng có khả năng tự động thực hiện những tác vụ hữu ích cho con người và điển hình nhất là trong lĩnh vực công nghệ máy tính.

1. *Nhược điểm*

* Giao diện Winform còn đơn giản
* Chưa tối ưu được điểm chính xác 100%
* Chưa tăng giới hạn số hình ảnh lên 10.000

1. *Phát triển*

Trong tương lai gần, với dự án này ta có thể phát triển trên App di động, tạo ra một mô hình mới độc quyền, huấn luyện trên mô hình tiên tiến hơn và mở rộng nghiên cứu bộ dữ liệu khác.

**TÀI LIỆU THAM KHẢO**

[1] <https://machinelearningcoban.com>

[2]<https://viblo.asia/p/ung-dung-convolutional-neural-network-trong-bai-toan-phan-loai-anh-4dbZNg8ylYM>

[3] <https://www.tensorflow.org>

[4] https://towardsdatascience.com/train-image-recognition-ai-with-5-lines-of-code-8ed0bdd8d9ba

[5] https://github.com/OlafenwaMoses/IdenProf/blob/master/README.md

[6] https://towardsdatascience.com/deep-learning-for-image-classification-why-its-challenging-where-we-ve-been-and-what-s-next-93b56948fcef

[7] <https://missinglink.ai/guides/computer-vision/neural-networks-image-recognition-methods-best-practices-applications/>

[8]https://topdev.vn/blog/thuat-toan-cnn-convolutional-neural-network/