

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



LUẬN VĂN TỐT NGHIỆP

**NGHIÊN CỨU KỸ THUẬT HỌC SÂU
VÀO NHẬN DẠNG CÁC ĐỐI TƯỢNG
GIAO THÔNG**

Người hướng dẫn: **TS PHẠM VĂN HUY**

Người thực hiện: **HOÀNG MINH TRUNG**

MSSV : 51303198

Lớp : 13050301

Khoá : 17

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2018

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



LUẬN VĂN TỐT NGHIỆP

**NGHIÊN CỨU KỸ THUẬT HỌC SÂU
VÀO NHẬN DẠNG CÁC ĐỐI TƯỢNG
GIAO THÔNG**

Người hướng dẫn: **TS PHẠM VĂN HUY**

Người thực hiện: **HOÀNG MINH TRUNG**

MSSV : 51303198

Lớp : 13050301

Khoá : 17

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2018

LỜI CẢM ƠN

Em xin chân thành cảm ơn Tiến Sĩ Phạm Văn Huy đã hướng dẫn em trong đề tài luận văn này. Thầy đã cho em hiểu rất nhiều kiến thức trong mảng xử lý ảnh, kỹ thuật học sâu, một số tài liệu và giải pháp liên quan đến vấn đề coding. Có những yêu cầu và những lời giải thích của thầy đề tài luận văn này hoàn thiện hơn. Một lần nữa em xin chân thành cảm ơn thầy.

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của TS Phạm Văn Huy. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận văn của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày tháng năm

Tác giả

(ký tên và ghi rõ họ tên)

Hoàng Minh Trung

TÓM TẮT

Trình bày tóm tắt vấn đề nghiên cứu, các hướng tiếp cận, cách giải quyết vấn đề và một số kết quả đạt được, những phát hiện cơ bản trong vòng 1 -2 trang.

MỤC LỤC

CHƯƠNG 1 – MỞ ĐẦU	6
1. Trí tuệ nhân tạo (AI – Artificial Intelligence)	6
2. Học máy (Machine Learning)	6
3. Học sâu (Deep Learning)	7
CHƯƠNG 2 – ARTIFICIAL NEURAL NETWORK	8
1. Khái niệm	8
2. Thành phần của mạng nơron	8
2.1. Tế bào thần kinh (Neurons)	8
2.2. Sự kết nối và trọng số (Connections and weights)	8
2.3. Chức năng lan truyền (Propagation function)	8
2.4. Quy luật học (Learning rule)	8
3. Layer	9
4. Các loại mô hình học	11
4.1. Học không giám sát	11
4.2. Học có giám sát	11
5. Perceptron	11
5.1. Perceptron	11
5.2. Multilayer Perceptron (MLP)	11
6. Linear regression	11
7. Activation function	11
8. Giải thuật Backpropagation hay còn gọi là Gradient descent	12
CHƯƠNG 3 – CONVOLUTIONAL NEURAL NETWORK	13
1. Khái niệm	13
2. Convolutional layer	13

3. Pooling layer	14
4. Full connected layer	14
5. Weights	14
CHƯƠNG 4 – TENSORFLOW API	15
1. Giới thiệu.....	15
2. Hỗ trợ ngôn ngữ	15
3. Các kiểu dữ liệu cơ bản của Tensorflow trên Python	15
3.1. <i>Rank</i>	16
3.2. <i>Shape</i>	16
3.3. <i>Type</i>	17
3.4. <i>Cơ chế hoạt động của Tensorflow</i>	17

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CÁC KÝ HIỆU

CÁC CHỮ VIẾT TẮT

ML: Machine Learning

AI: Artificial Intelligence

DL: Deep learning

DANH MỤC HÌNH

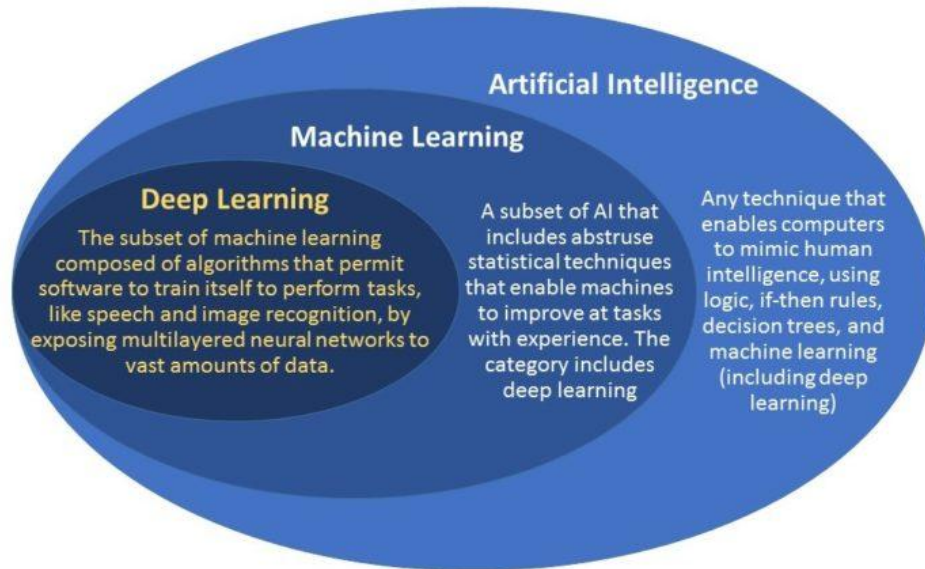
Hình 1 - Tổng quan về AI – ML - DL (Sưu tầm: techtalk)	6
Hình 2 - Layer trong <i>Neural Network</i>	9
Hình 3 - Các loại layer trong <i>Neural Network</i>	10
Hình 4 - Các tiến trình trong mô hình CNN(Sưu tầm –github.io)	13

DANH MỤC BẢNG

Bảng 1 – Rank trong Tensorflow	16
Bảng 2 – Shape trong Tensorflow	16
Bảng 3 – Kiểu dữ liệu trong Tensorflow	17

CHƯƠNG 1 – MỞ ĐẦU

- Dưới đây là bức tranh tổng quan về Trí tuệ nhân tạo, học máy và học sâu:



Hình 1 - Tổng quan về AI – ML - DL (Sưu tầm: techtalk)

1. Trí tuệ nhân tạo (AI – Artificial Intelligence)

- Trí tuệ nhân tạo là hành vi thông minh của một máy tính và hơn nữa là nó có thể có trí thông minh của con người và những động vật khác.
- Nói cách khác AI là cho máy tính bắt chước một chức năng nhận thức nào đó của con người và từ đó nó tự nhận thức được rồi đưa ra cách giải quyết vấn đề, việc này cũng giống như học và giải quyết vấn đề của con người.

2. Học máy (Machine Learning)

- Học máy là một lĩnh vực phát triển của AI, liên quan đến việc phát triển các kỹ thuật học dành cho máy tính
- Học máy là một phương pháp để tạo ra các chương trình máy tính bằng việc phân tích các tập dữ liệu: hình ảnh, văn bản,...
- Học máy tập trung vào các giải thuật phức tạp để thực thi trong việc tính toán, phân tích dữ liệu.

3. Học sâu (Deep Learning)

- Học sâu là một rẽ nhánh hẹp của học máy, nó không tập trung vào giải thuật như học máy mà nó đòi hỏi về độ lớn dữ liệu.
- Học sâu là một quá trình sử dụng thuật toán để học dữ liệu và biểu diễn dữ liệu theo từng layer theo mục đích nào đó

CHƯƠNG 2 – ARTIFICIAL NEURAL NETWORK

1. Khái niệm

- **Artificial neural network(ANN) – Mạng nơron nhân tạo** là mô hình tính toán dựa trên cấu trúc và chức năng của mạng nơron sinh học.
- ANN giống như bộ não con người, được học thông qua huấn luyện (training) và có khả năng lưu giữ lại những gì đã học (tri thức) và sử dụng những tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).
- Một ANN được dựa trên một bộ sưu tập của các đơn vị kết nối hay còn gọi là nơron nhân tạo. Các ANN này với ANN khác có thể truyền tín hiệu cho nhau

2. Thành phần của mạng nơron

2.1. Tế bào thần kinh(Neurons)

- Nó một PE(Processing Element) tượng trưng nó một nút trong mạng nơron

2.2. Sự kết nối và trọng số(Connections and weights)

- Mạng nơron được hình thành dựa trên nhiều connection kết nối lại với nhau và giữa 2 connection mà được liên kết với nhau thì luôn có trọng số.

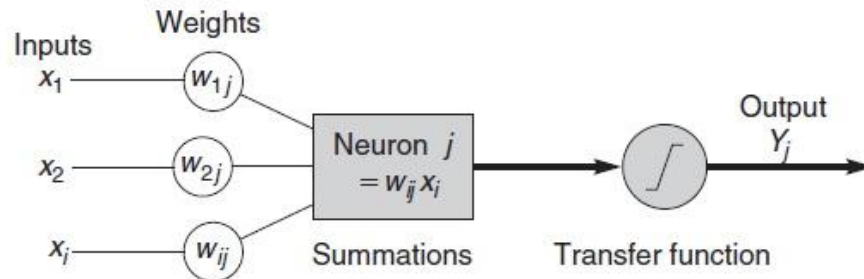
2.3. Chức năng lan truyền(Propagation function)

- Propagation function là cách tính input của connection này từ một output của connection khác

2.4. Quy luật học (Learning rule)

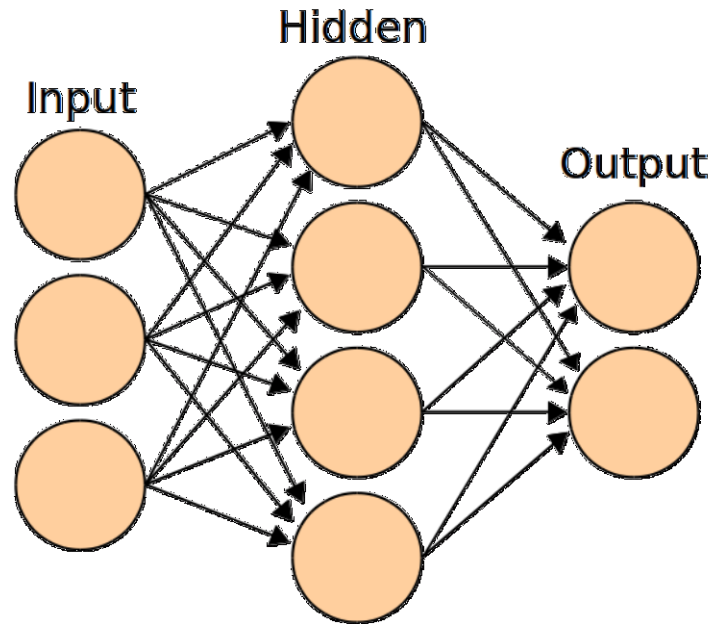
- Là một giải thuật dùng để tính toán từ một giá trị input đầu vào để cho ra một giá trị output hiệu quả
- Quá trình này chủ yếu là chỉnh sửa các trọng số và ngưỡng của các biến

3. Layer



Hình 2 - Layer trong *Neural Network*

- **Input layer:** Mỗi Input tương ứng với 1 thuộc tính (attribute) của dữ liệu (patterns)
- **Output layer:** Kết quả của một ANN là một giải pháp cho một vấn đề.
- **Hidden layer:** gồm các Nơron nhận dữ liệu input từ các Nơron ở lớp (Layer) trước đó và chuyển đổi các input này cho các lớp xử lý tiếp theo.
- **Connection Weights** (Trọng số liên kết) : Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng (độ tin tưởng) 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 dữ liệu đầu vào(input) để có được kết quả mong muốn.



Hình 3 - Các loại layer trong *Neural Network*

- ***Summation Function*** (Hàm tổng): Tính tổng trọng số của tất cả các input được đưa vào mỗi Nơron (phần tử xử lý PE). Hàm tổng của một Nơron đối với n input được tính theo công thức sau:

$$Y = \sum_{i=1}^n X_i W_i$$

- ***Transfer Function*** (Hàm chuyển đổi): Hàm tổng (Summation Function) của một Nơron cho biết khả năng kích hoạt (Activation) của Nơron đó còn gọi là kích hoạt bên trong (internal activation). Các Nơron này có thể sinh ra một output hoặc không trong ANN (nói cách khác rằng có thể output của 1 Nơron có thể được chuyển đến layer tiếp trong mạng Nơron hoặc không). Mối quan hệ giữa Internal Activation và kết quả (output) được thể hiện bằng hàm chuyển đổi (Transfer Function).

4. Các loại mô hình học

4.1. Học không giám sát

- Là một phương pháp của ngành học máy nhằm tìm ra một mô hình mà phù hợp với các quan sát.

4.2. Học có giám sát

- Là một kĩ thuật của ngành học máy để xây dựng một hàm (function) từ dữ liệu huấn luyện.
- Học có giám sát có thể tạo ra hai loại mô hình. Phổ biến nhất, học có giám sát tạo ra một mô hình toàn cục (global model) để ánh xạ đối tượng đầu vào đến đầu ra mong muốn. Tuy nhiên, trong một số trường hợp, việc ánh xạ được thực hiện dưới dạng một tập các mô hình cục bộ.

5. Perceptron

5.1. Perceptron

- Perceptron là một thuật toán Classification cho trường hợp đơn giản nhất: chỉ có hai lớp hay được gọi là binary classification và cũng chỉ hoạt động được trong một trường hợp rất cụ thể.

5.2. Multilayer Perceptron (MLP)

- Một multilayer perceptron (MLP) là việc implemnet nhiều layers perceptrons được kết nối liên kề nhau.
- MLP này giúp ích trong những hàm phi tuyến tính mà *perceptron* đơn không thể thực hiện.

6. Linear regression

- Dùng để phân chia 2 mặt phẳng thuộc 2 nhóm đối tượng khác nhau mà ít mất mát nhất

7. Activation function

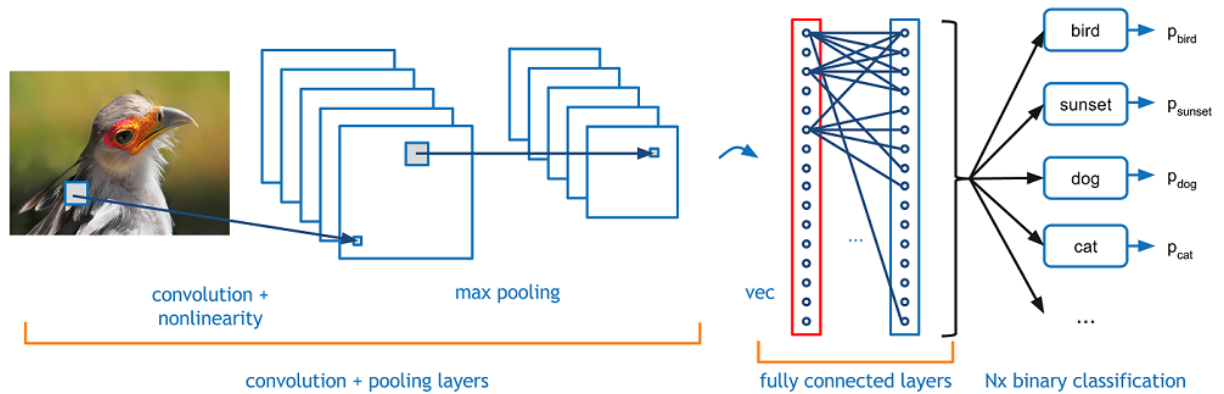
- Activation function là hàm nhận vector đầu vào, sau đó biến đổi để trả về vector đầu ra.

- Các loại activation function: *sigmoid function*, *ReLU*s, *Softmax function*

8. Giải thuật Backpropagation hay còn gọi là Gradient descent

- Gradient descent là một kỹ thuật dùng để tối ưu hóa hoặc cải tiến mô hình dự đoán trong **Neural Network** hay nói cách khác là giảm sự dự đoán sai cho đến mức tối thiểu.

CHƯƠNG 3 – CONVOLUTIONAL NEURAL NETWORK



Hình 4 - Các tiến trình trong mô hình CNN(Sưu tầm –github.io)

1. Khái niệm

- **Convolutional neural network(CNNs hay ConvNet) – Mạng nơron tích chập** là một **Deep Neural Network** thường được dùng trong lĩnh vực phân tích hình ảnh thị giác.
- CNN sử dụng một biến thể của **Multilayer Perceptron** nhằm giảm tối đa thời gian của giai đoạn tiền xử lý ảnh.
- Quá trình tiền xử lý ảnh của CNN tương đối ít hơn những thuật toán phân loại hình ảnh khác nó chủ yếu sử dụng các thuật toán bộ lọc truyền thống mà được viết bằng tay.

2. Convolutional layer

- Đây là một tầng áp dụng Convolutional để tính toán giá trị đầu vào cho tầng tiếp theo.
- Mỗi nơron Convolutional chỉ xử lý dữ liệu theo lĩnh vực tiếp nhận của nó
- Giải pháp của Convolutional là nó giải quyết các vấn đề gradients biến dạng hoặc biến mất trong việc huấn luyện các mạng nơ-ron đa lớp truyền thống với nhiều lớp bằng cách sử dụng phương pháp *Backpropagation*.

3. Pooling layer

- Max pooling là cách phổ biến được dùng ở layer này. Nó dùng để giảm kích thước của ma trận mà Convolutional Layer cung cấp

4. Full connected layer

- Fully connected kết nối mỗi nơ-ron trong một lớp với mỗi nơ-ron trong một lớp khác. Về nguyên tắc, nó giống như mạng nơ-ron đa lớp truyền thống.

5. Weights

- CNN có thể chia sẻ Weights trên những Convolutional layers để giảm bộ nhớ và cải tiến hiệu suất.

CHƯƠNG 4 – TENSORFLOW API

1. Giới thiệu

- **Tensorflow** là một thư viện phần mềm mã nguồn mở để tính toán số bằng cách sử dụng các biểu đồ dòng chảy dữ liệu.
- **Tensorflow** có kiến trúc linh hoạt cho phép bạn triển khai tính toán cho một hoặc nhiều CPU hoặc GPU trong một máy tính để bàn, máy chủ hoặc thiết bị di động với một API.
- **Tensorflow** được phát triển bởi các kỹ sư của Google và được dùng cho mục đích của việc tiến hành nghiên cứu học máy và Deep Neural Networks của họ.

2. Hỗ trợ ngôn ngữ

- C++, Python, Java, Go
- Về bản chất bên trong core thì Tensorflow API được implemented dựa trên ngôn ngữ C, C++.

3. Các kiểu dữ liệu cơ bản của Tensorflow trên Python

- **Tensor** là khái niệm cơ bản nhất trong TensorFlow.
- **Tensor** là cấu trúc dữ liệu được sử dụng trong toàn TensorFlow, đại diện cho tất cả các loại dữ liệu. Hay nói cách khác là tất cả các loại dữ liệu đều là tensor.
- Việc trao đổi dữ liệu trong quá trình xử lý chỉ thông qua tensor.
- Nói đơn giản hơn thì tensor là mảng n chiều hay list cộng thêm 1 vài thứ thú vị khác.
- Tensor có 3 thuộc tính là *Rank*, *Shape*, *Type*.

3.1. Rank

Rank	Loại	Ví dụ
0	Scalar	$s = 500$
1	Vector	$v = [1, 2, 3, 4, 5]$
2	Matrix	$m = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]$
3	3-Tensor	$t = [[[1], [2], [3]], [[4], [5], [6]], [[7], [8], [9]]]$
n	n-Tensor	$t = \text{mảng } n \text{ chiều}$

Bảng 1 – Rank trong Tensorflow

3.2. Shape

Rank	Shape	Số chiều	Ví dụ
0	$[]$	0	Scalar
1	$[D0]$	1	Vector có độ dài $[5]$
2	$[D0, D1]$	2	Ma trận 2 chiều $[5, 5]$
3	$[D0, D1, D2]$	3	Ma trận 3 chiều $[5, 5, 5]$
n	$[D0, D1, D2, \dots, D_{n-1}, D_n]$	4	Ma trận n chiều

Bảng 2 – Shape trong Tensorflow

3.3. Type

- Dưới đây là một số dạng dữ liệu thường dùng trong tensorflow.

Loại dữ liệu	Kiểu	Mô tả
DT_FLOAT	tf.float32	Kiểu số float 32 bit
DT_DOUBLE	tf.float64	Kiểu số float 64 bit
DT_INT8	tf.int8	Kiểu số int 8 bit
DT_INT16	tf.int16	Kiểu số int 16 bit
DT_INT32	tf.int32	Kiểu số int 32 bit
DT_UINT8	tf.uint8	Kiểu số int 8 bit không âm
DT_STRING	tf.string	Kiểu chuỗi nhưng trong tensorflow nó một mảng byte
DT_BOOL	tf.bool	Kiểu đúng sai

Bảng 3 – Kiểu dữ liệu trong Tensorflow

3.4. Cơ chế hoạt động của Tensorflow

- Vì tensorflow dùng graph để chạy nên run tensorflow có một chút đặc biệt là mỗi khi bạn muốn run cái gì đó bạn phải khai báo Session.

- Đa số các dữ liệu một muốn chạy trên tensor phải được cấp bộ nhớ Variable hoặc Placeholder

- Dưới đây là một ví dụ cơ bản về tensor:

```
init = tf.initialize_all_variables()
sess = tf.Session()
sess.run(init)
hello = tf.constant('Hello, TensorFlow!')
sess.run(hello)
```

Kết quả là: Hello, TensorFlow!