**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**----------🙡🕮🙣----------**



**BÀI TẬP LỚN**

**KHO DỮ LIỆU VÀ CÁC PHƯƠNG PHÁP KHAI PHÁ**

**ĐỀ TÀI: NHẬN DIỆN CHÓ MÈO SỬ DỤNG THUẬT TOÁN SVM VÀ NGÔN NGỮ PYTHON**

**Giáo viên hướng dẫn : TS. Trần Chí Kiên**

**Lớp : ĐH - KTPM 1 - K11**

**Nhóm sinh viên : Nhóm 10**

**Sinh viên thực hiện :**

|  |  |
| --- | --- |
| **Mã sinh viên** | **Họ và tên** |
| 1141360031 | Dương Thái Sơn |
| 1141360001 | Nguyễn Phú Nghĩa |

**Hà Nội - Năm 2020**

**MỤC LỤC**

[LỜI MỞ ĐẦU 4](#_Toc57986047)

[CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 5](#_Toc57986048)

[1.1. Tên đề tài 5](#_Toc57986049)

[1.2. Mục đích của đề tài 5](#_Toc57986050)

[1.3. Đối tượng ứng dụng và phạm vi nghiên cứu của đề tài 5](#_Toc57986051)

[1.4. Ý nghĩa 5](#_Toc57986052)

[CHƯƠNG II: TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU 6](#_Toc57986053)

[2.1. Các khái niệm cơ bản 6](#_Toc57986054)

[2.2. Các nhiệm vụ và công việc chính của khai phá dữ liệu 8](#_Toc57986055)

[2.3. Một số ứng dụng của khai phá dữ liệu 10](#_Toc57986056)

[2.4. Thách thức 13](#_Toc57986057)

[CHƯƠNG III: TỔNG QUAN VỀ NHẬN DẠNG 15](#_Toc57986058)

[3.1. Giới thiệu 15](#_Toc57986059)

[3.2. Quy trình 16](#_Toc57986060)

[3.3. Một số ứng dụng về nhận dạng 17](#_Toc57986061)

[CHƯƠNG IV: CÀI ĐẶT CHƯƠNG TRÌNH 19](#_Toc57986062)

[4.1. Thuật toán SVM 19](#_Toc57986063)

[4.1.1. Định nghĩa 19](#_Toc57986064)

[4.1.2. SVM cho 2 lớp dữ liệu 20](#_Toc57986065)

[4.1.3. Kernel SVM 24](#_Toc57986066)

[4.1.4. Ưu điểm của SVM 25](#_Toc57986067)

[4.1.5. Nhược điểm của SVM 26](#_Toc57986068)

[4.2. Tensorflow 26](#_Toc57986069)

[4.2.1. Định nghĩa 26](#_Toc57986070)

[4.2.2. Các thành phần chính 27](#_Toc57986071)

[4.2.3. Ưu điểm 27](#_Toc57986072)

[4.3. Mạng Nơ-ron (Neural Network) 28](#_Toc57986073)

[4.3.1. Định nghĩa 28](#_Toc57986074)

[4.3.2. Các đặc điểm của mạng nơ-ron 28](#_Toc57986075)

[4.3.3. Ứng dụng của mạng Nơ-ron 29](#_Toc57986076)

[4.4. Cài đặt chương trình 30](#_Toc57986077)

[4.4.1. Dataset của chó và mèo 30](#_Toc57986078)

[4.4.2. Train Data 31](#_Toc57986079)

[4.4.2. Predict 31](#_Toc57986080)

[LỜI CẢM ƠN 32](#_Toc57986081)

[TÀI LIỆU THAM KHẢO 33](#_Toc57986082)

**DANH MỤC HÌNH VẼ**

[Hình 1. Các mặt phân cách hai class linearly separable 21](#_Toc57926379)

[Hình 2. Margin của hai class là bằng nhau và lớn nhất có thể 22](#_Toc57926380)

[Hình 3. Soft margin 23](#_Toc57926381)

[Hình 4. Dữ liệu không tuyến tính được ánh xạ sang một không gian khác 24](#_Toc57926382)

[Hình 5. Dataset mèo 30](#_Toc57926383)

[Hình 6. Dataset chó 30](#_Toc57926384)

# LỜI MỞ ĐẦU

Hiện nay, cùng với sự phát triển của khoa học và xã hội, vấn đề an ninh, bào mật đang được yêu cầu khắt khe tại mọi quốc gia trên thế giới. Các hệ thống nhận dạng con người, đồ vật,… được ra đời và phát triển với độ tin cậy ngày càng cao. Với cách tiếp cận đối tượng nhận dạng theo phương pháp này, chúng ta có thể thu nhận được nhiều thông tin từ đối tượng hơn, mà không cần tác động nhiều đến đối tượng cũng vẫn đảm bảo tính chính xác, an toàn, thuận tiện.

Một trong những bài toán được xã hội quan tâm hiện nay là nhận dạng con vật. Chúng ta có thể liệt kê ra một số thuật toán cụ thể để giải quyết vấn đề nhận dạng như: PCA, LDA, ICA, SVM, EP, EBGM,… Tuy nhiên vì điều kiện thời gian nên nhóm em chỉ tập trung vào tìm hiểu và làm một ứng dụng nhỏ sử dụng thuật toán thông dụng đó là SVM (Support Vector Machines). Đồng thời nhóm bọn em cũng muốn có những hiểu biết sâu hơn về lý thuyết nhận dạng nên nhóm em đã chọn đề tài: **”Nhận diện chó mèo sử dụng thuật toán SVM và ngôn ngữ Python”.**

Do kiến thức bản thân còn hạn hẹp cũng như chưa có kinh nghiệm làm một chương trình có tính ứng dụng thực tế cao nên trong quá trình thực hiện nhóm em còn phạm phải nhiều sai sót cũng như chương trình có thể còn tồn tại lỗi không mong muốn. Rất mong nhận được góp ý của thầy để chúng em hoàn thiện hơn và rút kinh nghiệm cho những lần làm sau tốt hơn.

**Chúng em xin chân thành cảm ơn!**

**Sinh viên nhóm thực hiện: Nhóm 10**

# TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Tên đề tài

Nhận diện chó mèo sử dụng thuật toán SVM và ngôn ngữ Python

## 1.2. Mục đích của đề tài

Giúp người dùng dễ dàng phân biệt được chó và mèo dựa vào hình ảnh.

## 1.3. Đối tượng ứng dụng và phạm vi nghiên cứu của đề tài

* Phạm vi nghiên cứu
* Sản phẩm của đề tài được xây dựng trên ngôn ngữ Python, sử dụng mã nguồn mở Tensorflow và Keras.
* **Nhận diện chó và mèo**

Người dùng chọn ảnh chó hoặc mèo từ máy và ấn nút nhận diện, kết quả trả về sẽ xác định ảnh vừa nhập là chó hay là mèo.

**1.4. Ý nghĩa**

Ứng dụng được xây dựng nhằm mục đích nhận diện chó và mèo một cách nhanh nhất.

# TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU

## 2.1. Các khái niệm cơ bản

Data Mining là một trong những thuật ngữ mới xuất hiện đầu thế kỷ 21, nó là hệ quả của sự bùng nổ Internet đạt tới đỉnh điểm. Theo một công bố của Intel vào tháng 9 năm 2013 cứ 11 giây trôi qua chúng ta có thêm 1 Petabybe dữ liệu, nó tương đương với một video chất lượng HD dài… 13 năm. Và để khai phá, trích xuất nó Data Mining ra đời, dưới đây là khái niệm Data Mining là gì.

Khai phá dữ liệu là quá trình xác định các mẫu tiềm ẩn có tính hợp lệ, mới lạ, có ích và có thể hiểu được trong một khối dữ liệu rất lớn.

Data Mining là khai phá dữ liệu, Data Mining là quá trình khai phá, trích xuất, khai thác và sử dụng những dữ liệu có giá trị tiềm ẩn từ bên trong lượng lớn dữ liệu được lưu trữ trong các cơ sở dữ liệu (CSDL), kho dữ liệu, trung tâm dữ liệu… lớn hơn là Big Data dựa trên kĩ thuật như mạng nơ ron, lí thuyết tập thô, tập mờ, biểu diễn tri thức… Đây là một công đoạn trong hoạt động “làm sạch” dữ liệu.

Data Mining phân tích các mối quan hệ và các mẫu trong các dữ liệu được lưu trữ dựa trên các truy vấn của người dùng. Khai thác dữ liệu liên quan đến những nhiệm vụ như sau:

* Association (Kết hợp): Tìm mối quan hệ giữa các biến.

Ví dụ như trong một cửa hàng bán lẻ, có thể xác định sản phẩm được mua cùng với nhau thường xuyên và thông tin này có thể được sử dụng để tiếp thị các sản phẩm này.

* Clustering (Phân cụm): Xác định mối quan hệ hợp lý trong các sản phẩm và nhóm chúng lại với nhau.

Ví dụ như trong một cửa hàng bán lẻ, kem đánh răng và bàn chải đánh răng có thể được nhóm lại.

* Classifying (Phân loại):

Liên quan đến việc áp dụng một mô hình được biết đến với các dữ liệu mới.

Để dễ hình dung rõ hơn Data Mining là gì thì có thể hiểu đơn giản nó chính là một phần của quá trình trích xuất những dữ liệu có giá trị tốt, loại bỏ dữ liệu giá trị xấu trong bộn bề thông tin trên Internet và các nguồn dữ liệu đang có.

Data Mining là quá trình: khai phá, trích xuất, khai thác và sử dụng dữ liệu có giá trị trong các CSDL đang có (kể cả Big Data)  
- Đây là công đoạn đưa ra dữ liệu đã được “làm sạch”  
- Dựa trên kĩ thuật như mạng nơ ron, lí thuyết tập thô, tập mờ, biểu diễn tri thức…

Quá trình chọn lọc dữ liệu của Data Mining dựa trên các phương pháp: Phân loại (Classification), Phân nhóm (Clustering), Tổng hợp (Summarization), Mô hình ràng buộc (Dependency modeling), Hồi qui (Regression), Dò tìm biến đổi và độ lệch (Change and Deviation Dectection).

Để có nguồn Data Mining chúng ta cần thực hiện các công đoạn: Làm sạch dữ liệu (Data Cleaning); Tích hợp dữ liệu (Data Integration); Chọn dữ liệu (Data Selection); Chuyển đổi dữ liệu (Data Transformation). Công đoạn sau của Data Mining chính là: Đánh giá mẫu (Pattern Evaluation); Trình diễn dữ liệu (Knowlegde Presentation). Chúng tôi sẽ sơ đồ hóa để bạn đọc dễ hiểu hơn Data Mining là gì, Data Mining là công đoạn thứ bao nhiêu trong quá trình làm sạch dữ liệu.

Data Cleaning => Data Integration => Data Selection => Data Transformation => Data Mining => Pattern Evaluation => Knowlegde Presentation.

Như vậy có thể thấy Data Mining chính là một “mắt xích” trong quá trình làm sạch dữ liệu trong các cơ sở dữ liệu. Nếu bạn đọc đang sử dụng nguồn dữ liệu lớn – Big Data thì quá trình khai phá chúng quả không dễ dàng gì bạn có thể mất nhiều thời gian hơn, nhiều nhân lực, chi phí cho hoạt động Data Mining trong Big Data của mình.

Tuy nhiên trường hợp một cá nhân sở hữu Big Data là khá hiếm nó chỉ thuộc về một doanh nghiệp tầm cỡ, rộng hơn là quy mô toàn cầu, dữ liệu toàn cầu… để giải thích khái niệm Big Data có nghĩa là gì bạn đọc có thể tham khảo bài viết Big Data là gì mà chúng tôi kỳ công nghiện cứu, thực hiện mới đây.

Khai phá dữ liệu là quá trình xác định các mẫu tiềm ẩn có tính hợp lệ, mới lạ, có ích và có thể hiểu được trong một khối dữ liệu rất lớn.

Khai phá dữ liệu là một bước của quá trình [khai thác tri thức](https://vi.wikipedia.org/w/index.php?title=Khai_th%C3%A1c_tri_th%E1%BB%A9c&action=edit&redlink=1) (Knowledge Discovery Process), bao gồm:

* Xác định vấn đề và không gian dữ liệu để giải quyết vấn đề (Problem understanding and data understanding).
* Chuẩn bị dữ liệu (Data preparation), bao gồm các quá trình làm sạch dữ liệu (data cleaning), tích hợp dữ liệu (data integration), chọn dữ liệu (data selection), biến đổi dữ liệu (data transformation).
* Khai thác dữ liệu (Data mining): xác định nhiệm vụ khai thác dữ liệu và lựa chọn kỹ thuật khai thác dữ liệu. Kết quả cho ta một nguồn tri thức thô.
* Đánh giá (Evaluation): dựa trên một số tiêu chí tiến hành kiểm tra và lọc nguồn tri thức thu được.
* Triển khai (Deployment).

Quá trình khai thác tri thức không chỉ là một quá trình tuần tự từ bước đầu tiên đến bước cuối cùng mà là một quá trình lặp và có quay trở lại các bước đã qua.

## 2.2. Các nhiệm vụ và công việc chính của khai phá dữ liệu

* **Nhiệm vụ cơ bản nhất của khai phá dữ liệu là:**

Phân cụm, phân loại, phân nhóm, phân lớp: quá trình này thường được thực hiện một cách tự động.

- Khai phá luật kết hợp: Nhiệm vụ là phát hiện ra những mối quan hệ giống nhau của các bản ghi giao dịch. Luật kết hợp X=>Y có dạng tổng quát là: Nếu một giao dịch đã sở hữu các tính chất X thì đồng thời nó cũng sở hữu các tính chất Y, ở một mức độ nào đó. Khai phá luật kết hợp được hiểu theo nghĩa: Biết trước các tính chât X, vậy các tính chất Y là những tính chất nào?

- Lập mô hình dự báo: bao gồm hai nhiệm vụ: Hoặc là phân nhóm dữ liệu vào một hay nhiều lớp dữ liệu đã xác định từ trước, hoặc là sử dụng các trường đã cho trong một cơ sở dữ liệu để dự báo sự xuất hiện (hoặc không xuất hiện) của các trường hợp khác.

- Phân tích đối tượng ngoài cuộc: Một cơ sở dữ liệu có thể có thể chứa các đối tượng không tuân theo mô hình dữ liệu. Các đối tượng dữ liệu như vậy gọi là các đối tượng ngoài cuộc.

- Phân tích sự tiến hóa: Phân tích sự tiến hóa thực hiện việc mô tả và mô hình hóa các quy luật hay khuynh hướng của những đối tượng mà ứng xử của chúng thay đổi theo thời gian.

* **Kỹ thuật khai phá dữ liệu gồm các nhóm chính sau**:

- Kỹ thuật khai phá dữ liệu mô tả: Có nhiệm vụ mô tả về các tính chất hoặc các đặc tính chung của dữ liệu trong cơ sở dữ liệu hiện có. Các kỹ thuật này gồm có: Phân cụm (clustering), tóm tắt (summerization), trực quan hóa (visualiztation), phân tích sự phát triển và độ lệch (evolution and deviation analyst), phân tích luật kết hợp (association rules) …

- Kỹ thuật khai phá dữ liệu dự đoán: Có nhiệm vụ đưa ra các dự đoán dựa vào các suy diễn trên dữ liệu hiện thời. Các kỹ thuật này gồm có: Phân lớp (classification), hồi quy 7 (regession)… Tuy nhiên, chỉ có một số phương pháp thông dụng nhất là: Phân cụm dữ liệu, phân lớp dữ liệu, phương pháp hồi quy và khai phá luật kết hợp.

- Phân cụm dữ liệu: Mục tiêu chính của phương pháp phân cụm dữ liệu là nhóm các đối tượng tương tự nhau trong tập dữ liệu vào các cụm sao cho các đối tượng thuộc cùng một lớp là tương đồng còn các đối tượng thuộc các cụm khác nhau sẽ không tương đồng.

- Phân lớp dữ liệu: Mục tiêu của phương pháp phân lớp dữ liệu là dự đoán nhãn lớp cho các mẫu dữ liệu. Quá trình phân lớp dữ liệu thường gồm hai bước: Xây dựng mô hình và sử dụng mô hình để phân lớp dữ liệu.

Bước 1: Xây dựng mô hình: Một mô hình sẽ được xây dựng dựa trên việc phân tích các mẫu dữ liệu sẵn có. Mỗi mẫu tương ứng với một lớp, được quyết định bởi một thuộc tính gọi là thuộc tính lớp. Các lớp dữ liệu này còn được gọi là lớp dữ liệu huấn luyện (training data set). Các nhãn lớp của tập dữ liệu huấn luyện đều phải được xác định trước khi xây dựng mô hình.

Bước 2: Sử dụng mô hình để phân lớp dữ liệu. Trước hết, chúng ta phải tính độ chính xác của mô hình. Nếu độ chính xác là chấp nhận được, mô hình sẽ được sử dụng để dự đoán nhãn lớp cho các mẫu dữ liệu khác trong tương lai.

- Phương pháp hồi quy: Phương pháp hồi quy khác với phân lớp dữ liệu ở chỗ: Hồi quy dùng để dự đoán về các giá trị liên tục còn phân lớp dữ liệu chỉ dùng để dự đoán về các giá trị rời rạc.

- Khai phá luật kết hợp: Mục tiêu của phương pháp này là phát hiện và đưa ra các mối liên hệ giữa các giá trị dữ liệu trong cơ sở dữ liệu. Mẫu đầu ra của giải thuật khai phá dữ liệu là luật kết hợp tìm được.

Khai phá luật kết hợp được thực hiện qua hai bước:

Bước 1: Tìm tất cả các tập mục phổ biến, một tập mục phổ biến được xác định qua tính hỗ trợ và thỏa mãn độ hỗ trợ cực tiểu

Bước 2: Sinh ra các luật kết hợp mạnh từ tập mục phổ biến, các luật phải thỏa mãn độ hỗ trợ cực tiểu và độ tin cậy cực tiểu. Phương pháp này được sử dụng rất hiệu quả trong các lĩnh vực như maketing có chủ đích, phân tích quyết định, quản lý kinh doanh, phân tích giá thị trường …

## 2.3. Một số ứng dụng của khai phá dữ liệu

Kinh tế - ứng dụng trong kinh doanh, tài chính, tiếp thị bán hàng, bảo hiểm,thương mại, ngân hàng, … Đưa ra các bản báo cáo giàu thông tin; phân tích rủi ro trước khi đưa ra các chiến lược kinh doanh, sản xuất; phân loại khách hàng từ đó phân định thị trường, thị phần; …

* Phân tích dữ liệu tài chính (Financial Data Analysis)
* Công nghiệp bán lẻ (Retail Industry)
* Công nghiệp viễn thông (Telecommunication Industry)
* Phân tích dữ liệu sinh học (Biological Data Analysis)
* Phát hiện xâm nhập (Intrusion Detection)
* Một số ứng dụng trong khoa học (Scientific Applications)
* Web: các công cụ tìm kiếm.
* Công nghệ sinh học –  tìm ra các gen mới, cây con giống mới, …
* Khoa học: Thiên văn học– dự đoán đường đicác thiên thể, hành tinh, ….

**Phân tích dữ liệu tài chính (Financial Data Analysis)**

Dữ liệu tài chính trong ngân hàng và trong ngành tài chính nói chung thường đáng tin cậy và có chất lượng cao, tạo điều kiện cho khai phá dữ liệu. Dưới đây là một số ứng dụng điển hình trong khai phá dữ liệu tài chính:

* Dự đoán khả năng vay và thanh toán của khách hàng, phân tích chính sách tín dụng đối với khách hàng.
* Phân tích hành vi khách hàng (vay, gửi tiền)
* Phân loại và phân nhóm khách hàng mục tiêu cho tiếp thị tài chính
* Phát hiện các hoạt động rửa tiền và tội phạm tài chính khác

**Công nghiệp bán lẻ (Retail Industry)**

Khai phá dữ liệu có vai trò rất quan trọng trong ngành công nghiệp bán lẻ, do dữ liệu  thu thập từ lĩnh vực này rất lớn từ doanh số bán hàng, lịch sử mua hàng của khách hàng, vận chuyển hàng hóa, tiêu thụ và dịch vụ. Điều tự nhiên là khối lượng dữ liệu từ ngành công nghiệp này sẽ tiếp tục tăng lên nhanh chóng và dễ dàng thu thập bởi tính sẵn có trên môi trường Web. Ứng dụng khai phá dữ liệu trong ngành công nghiệp bán lẻ nhằm xây dựng mô hình giúp xác định xu hướng mua hàng của khách hàng, giúp doanh nghiệp cải thiện chất lượng sản phẩm dịch vụ nhằm nâng cao sự hài lòng của khách hàng và giữ chân khách hàng tốt. Dưới đây là một số ứng dụng của khai phá dữ liệu trong ngành công nghiệp bán lẻ:

* Khai phá dữ liệu trên kho dữ liệu khách hàng
* Phân tích đa chiều trên kho dữ liệu khách hàng về doanh số bán hàng, khách hàng, sản phẩm, thời gian và khu vực.
* Phân tích hiệu quả của các chiến dịch bán hàng, Marketing.
* Quản trị mối quan hệ khách hàng (CRM).
* Giới thiệu và tư vấn sản phẩm phù hợp cho khách hàng

**Công nghiệp viễn thông (Telecommunication Industry)**

Công nghiệp viễn thông là một trong những ngành công nghiệp mới nổi, cung cấp nhiều dịch vụ như trên điện thoại di động, Internet, truyền hình ảnh... Do sự phát triển mạnh của công nghệ máy tính và mạng máy tính, viễn thông đang phát triển với tốc độ rất lớn. Đây là lý do tại sao khai phá dữ liệu trở nên rất quan trọng trong lĩnh vực này.   
Khai phá dữ liệu trong ngành công nghiệp viễn thông giúp xác định các mô hình viễn thông, phát hiện các hoạt động gian lận trong viễn thông, sử dụng tốt hơn nguồn tài nguyên và cải thiện chất lượng dịch vụ viễn thông. Dưới đây là một số ứng dụng của khai phá dữ liệutrong ngành công nghiệp này:

* Phân tích dữ liệu đa chiều viễn thông.
* Xây dựng các mô hình phát hiện gian lận.
* Phát hiện bất thường trong giao dịch viễn thông.
* Phân tích hành vi sử dụng dịch vụ viễn thông của khách hàng
* Sử dụng các công cụ trực quan trong phân tích dữ liệu viễn thông

**Phân tích dữ liệu sinh học (Biological Data Analysis)**

Khai phá dữ liệu sinh học là một phần rất quan trọng của lĩnh vực Tin -Sinh học (Bioinformatics). Sau đây là một số ứng dụng của khai phá dữ liệu ứng dụng trong sinh học:

* Lập chỉ mục, tìm kiếm tương tự, bất thường trong cơ sở dữ liệu Gen.
* Xây dựng mô hình khai phá các mạng di truyền và cấu trúc của Gen, protein
* Xây dựng các công cụ trực quan trong phân tích dữ liệu di truyền.

**Phát hiện xâm nhập bất hợp pháp (Intrusion Detection)**

Xâm nhập bất hợp pháp là những hành động đe dọa tính toàn vẹn, bảo mật và tính sẵn sàng của tài nguyên mạng. Trong thế giới của kết nối, bảo mật đã trở thành vấn đề lớn đối với tồn tại của hệ thống. Với sự phát triển của internet và sự sẵn có của các công cụ, thủ thuật trợ giúp cho xâm nhập và tấn công mạng, yêu cầu kiểm soát truy cập bất hợp pháp là yếu tố rất quan trọng đảm bảo cho sự ổn định của hệ thống.

Dưới đây là một số ứng dụng của khai phá dữ liệu có thể được áp dụng để phát hiện xâm nhập:

* Phát triển các thuật toán khai phá dữ liệu để phát hiện xâm nhập.
* Phân tích kết hợp, tương quan và khác biệt để phát hiện xâm nhập.
* Phân tích dòng dữ liệu dữ liệu (Analysis of Stream data) để phát hiện bất thường.

## 2.4. Thách thức

10 vấn đề thách thức hàng đầu trong khai phá dữ liệu:

* Phát triển một lý thuyết thống nhất cho khai phá dữ liệu
* Xử lý dữ liệu với số chiều cao và dòng dữ liệu tốc độ cao
* Khai phá dữ liệu tuần tự và dữ liệu chuỗi thời gian
* Khai phá tri thức phức hợp từ dữ liệu phức hợp
* Khai phá dữ liệu trong các cấu hình mạng
* Khai phá dữ liệu phân tán và dữ liệu đa tác tử
* Khai phá dữ liệu sinh học và các vấn đề môi trường
* Các bài toán liên quan đến quy trình khai phá dữ liệu
* An ninh, bảo mật và toàn vẹn dữ liệu
* Khai phá dữ liệu không tĩnh, không cân bằng, chi phí lớn.

Ngoài ra, còn 1 số thách thức như:

- Cơ sở dữ liệu lớn: kích thước của cơ sở dữ liệu được nhận biết thông qua số lượng các mẫu tin, các thuộc tính (hay các biến) và các bảng, số lượng có thể là hàng trăm thuộc tính và bảng, hàng triệu các mẫu tin. Như vậy, kích thước của cơ sở dữ liệu tính bằng terabyte (1012 byte) đã bắt đầu xuất hiện. Dữ liệu với số chiều (tương ứng với thuộc tính khi biểu diễn qua không gian các mẫu dữ liệu) cao tạo nên sự gia tăng về kích thước của không gian tìm kiếm trong việc quy nạp mô hình, một sự bùng nổ về tổ hợp. Khi xây dựng mô hình chỉ một tập con trong cơ sở dữ liệu tham gia, vì vậy tính may rủi trong các thuật toán khai phá sẽ tìm được các mẫu không có giá trị trong trường hợp tổng quát. Một giải pháp cho vấn đề này là giảm bớt đáng kể số chiều của bài toán và sử dụng tri thức trước (prior knowledge) để nhận biết các biến ít liên quan.

- Vấn đề “quá khớp” (Over-fitting): Khi thuật toán khai phá tìm kiếm với các tham số tốt nhất cho một mô hình đặc biệt và một giới hạn của tập dữ liệu, mô hình ấy có thể “quá khớp” trên tập dữ liệu ấy nhưng lại thi hành không chính xác trên tập dữ liệu kiểm tra. Một giải pháp thường được sử dụng là thẩm định chéo.

- Thay đổi dữ liệu và tri thức: Dữ liệu là không tĩnh, dữ liệu thay đổi nhanh chóng có thể dẫn đến những mẫu đã khai phá trước đây không còn hiệu lực. Thêm vào đó, các biến đã được đo trong cơ sở dữ liệu ứng dụng đã bị thay đổi, bị xóa hoặc đã tăng lên với một độ đo mới. Điều này có thể được thực hiện bằng cách gia tăng các phương thức cập nhật các mẫu và xem xét các thay đổi như là một cơ hội cho việc khám phá bằng việc sử dụng nó để xử lý thích hợp việc tìm kiếm các mẫu chỉ với sự thay đổi.

- Dữ liệu thiếu và nhiễu: Đây là vấn đề rất được quan tâm trong khai phá dữ liệu, điều này thường dẫn đến việc dự đoán thiếu chính xác.

- Tích hợp với hệ thống: Hệ thống khai phá dữ liệu thực sự là hữu ích khi phải được tích hợp với cơ sở dữ liệu thông qua các giao diện như truy vấn, bảng tính và các công cụ trực quan khác. Hơn nữa, phải tạo ra một môi trường thuận lợi cho việc tương tác với người dùng.

# TỔNG QUAN VỀ NHẬN DẠNG

## 3.1. Giới thiệu

Nhận dạng mẫu (pattern recognition) là một ngành thuộc lĩnh vực học máy (machine learning). Nói cách khác, nó có thể được xem là việc "cần thực hiện một tác động vào dữ liệu thô mà tác động cụ thể là gì sẽ tùy vào loại của dữ liệu đó". Như vậy nó là một tập các phương pháp học có giám sát (supervised learning).

Nhận dạng mẫu nhằm mục đích phân loại dữ liệu (là các mẫu) dựa trên: hoặc là kiến thức tiên nghiệm (a priori) hoặc dựa vào thông tin thống kế được trích rút từ các mẫu có sẵn. Các mẫu cần phân loại thường được biểu diễn thành các nhóm của các dữ liệu đo đạc hay quan sát được, mỗi nhóm là một điểm ở trong một không gian đa chiều phù hợp. Đó là không gian của các đặc tính để dựa vào đó ta có thể phân loại.

Một hệ thống nhận dạng mẫu hoàn thiện gồm một thiết bị cảm nhận (sensor) để thu thập các quan sát cần cho việc phân loại hay miêu tả; một cơ chế trích chọn đặc trưng (feature extraction) để tính toán các thông tin dưới dạng số hay dạng tượng trưng (symbolic) từ các dữ liệu quan sát được; và một bộ phân loại (hay lược đồ mô tả) nhằm thực hiện công việc phân loại thực sự (hay miêu tả các quan sát đó) dựa vào các đặc tính đã được trích rút.

Việc phân loại (hay lược đồ mô tả) thường dựa vào sự có sẵn của một tập các mẫu mà đã được phân loại (hay miêu tả) sẵn. Tập các mẫu này được gọi là tập huấn luyện và chiến lược học nhằm phân loại mẫu vào một trong các lớp có sẵn được gọi là học có giám sát. Việc học cũng có thể là không có giám sát, theo nghĩa là hệ thống không được cung cấp các mẫu được đánh nhãn (phân loại) tiên nghiệm, mà nó phải tự đưa ra các lớp để phân loại dựa vào tính ổn định trong thống kê của các mẫu.

Việc phân loại (hay lược đồ mô tả) thường dùng một trong các hướng tiếp cận sau: thống kê (hay lý thuyết quyết định), cú pháp (hay cấu trúc). Nhận dạng mẫu dùng thống kê là dựa vào các đặc tính thống kê của các mẫu, chẳng hạn rằng các mẫu được tạo mởi một hệ thống xác xuất. Nhận dạng dùng cấu trúc là dựa vào tương quan cấu trúc giữa các mẫu.

Các ứng dụng phổ biến là nhận dạng tiếng nói tự động, phân loại chữ viết (ví dụ: những thư điện tử nào là spam/non-spam), nhận dạng các mã bưu điện viết tay trên các bao thư, hay hệ thống nhận dạng danh tính dựa vào mặt người. Ba ví dụ cuối tạo thành lãnh vực con phân tích ảnh của nhận dạng mẫu với đầu vào là các ảnh số.

## 3.2. Quy trình

Tùy từng đối tượng và mục đích nhận dạng mà mỗi hệ thống nhận dạng có thiết kế khác nhau. Tuy nhiên sơ đồ sau đây thường được sử dụng như một mẫu chung để thiết kế nhận dạng.

Hậu xử lý

Phân lớp, hồi quy, mô tả

Tiền xử lý

Thu nhận mẫu

Các bộ phận chức năng chính bao gồm:

* **Thu nhận mẫu:** có thể dưới nhiều dạng khác nhau như thu nhận tín hiệu, thu thập ảnh hoặc thu thập dữ liệu. Quá trình này được thực hiện bằng cách sử dụng các cảm biến, máy thu (camera, micro,...).
* **Tiền xử lý:** trong một vài trường hợp, các giá trị của các thuộc tính không được đưa trực tiếp vào các công đoạn sau. Thông thường chúng được chuẩn hóa (ví dụ đưa các giá trị về đoạn [0,1]) theo một cách nào đó. Đôi khi quá trình tiền xử lý còn có thể là các công việc làm sạch, làm giàu hoặc chuyển đổi định dạng dữ liệu hoặc trích chọn thuộc tính. Khâu trích chọn thuộc tính thường được áp dụng khi hệ thống phải đối mặt với lượng dữ liệu lớn, có nhiễu. Việc trích chọn những thông tin nào, trích chọn ra sao thường đã được nghiên cứu và thực hiện bởi các giải thuật cụ thể. Đôi khi, việc thực hiện trích chọn thuộc tính còn giúp nâng cao hiệu suất hệ thống.
* **Phân lớp, hồi quy, mô tả:** là những chức năng cơ bản của một hệ thống nhận dạng mà mục đích cuối cùng là phân loại các mẫu và nhận biết những đặc điểm chung của chúng.
* **Hậu xử lý:** đôi khi dữ liệu đầu ra của hệ thống nhận dạng không được sử dụng trực tiếp mà còn trải qua quá trình hậu xử lý, chẳng hạn như các thao tác giải mã.

## 3.3. Một số ứng dụng về nhận dạng

* **Nhận dạng biển số xe với openCV by step**

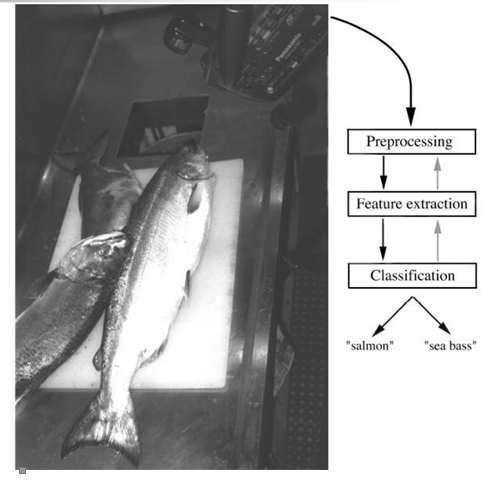
Một số biển số xe cần nhận dạng trên mạng:



Các bước thực hiện :

1. Nhận diện được vị trí của biển số xe trên image ( Object Localization)
2. Segmentation các kí tự trên biển số xe
3. Nhận dạng

* **Bài toán phân loại cá trong dây chuyền thành hai loại**



Các bước thực hiện:

1. Tiền xử lý ảnh cá nhận được từ camera.
2. Phân vùng đối tượng cá
3. Trích chọn đặc trưng: độ sáng, chiều dài, chiều rộng
4. Phân loại đối tượng cá

# CÀI ĐẶT CHƯƠNG TRÌNH

## 4.1. Thuật toán SVM

### ****4.1.1. Định nghĩa****

**Support Vector Machine**(SVM) là một thuật toán thuộc nhóm [Supervised Learning](https://1upnote.me/tags/supervised-learning/) (Học có giám sát) dùng để phân chia dữ liệu (Classification) thành các nhóm riêng biệt.

Support Vector Machines là một trong số những thuật toán phổ biến và được sử dụng nhiều nhất trong học máy trước khi mạng nơ ron nhân tạo trở lại với các mô hình deep learning. Nó được biết đến rộng rãi ngay từ khi mới được phát triển vào những năm 1990.

Mục tiêu của SVM là tìm ra một siêu phẳng trong không gian N chiều (ứng với N đặc trưng) chia dữ liệu thành hai phần tương ứng với lớp của chúng. Nói theo ngôn ngữ của đại số tuyển tính, siêu phẳng này phải có lề cực đại và phân chia hai bao lồi và cách đều chúng.

Trước khi đi sâu vào tìm hiểu thuật tuấn SVM, chúng ta xem lại một số kiến thức về hình học giải tích:

* Trong không gian 2 chiều, ta biết rằng khoảng cách từ một điểm có toạ độ **(x0,y0)** tới đường thẳng có phương trình **w1x+w2y+b=0** được xác định bởi:
* Trong không gian 3 chiều, ta biết rằng khoảng cách từ một điểm có toạ độ **(x0, y0, z0)** tới đường thẳng có phương trình **w1x+w2y+w3z+b=0** được xác định bởi:

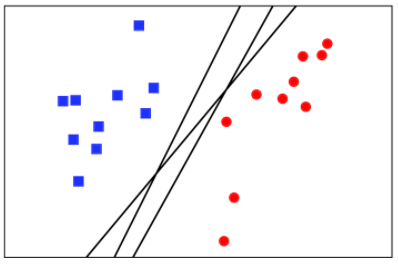
Hơn nữa, nếu ta bỏ dấu trị tuyệt đối ở tử số, chúng ta có thể xác định được điểm đó nằm về phía nào của đường thẳng hay mặt phẳng đang xét. Những điểm làm cho biểu thức trong dấu giá trị tuyệt đối mang dấu dương nằm về cùng 1 phía (tạm gọi đây là phía dương của đường thẳng), những điểm làm cho biểu thức trong dấu giá trị tuyệt đối mang dấu âm nằm về phía còn lại (gọi là phía âm). Những điểm nằm trên đường thẳng/măt phẳng sẽ làm cho tử số có giá trị bằng 0, tức khoảng cách bằng 0.

Việc này có thể được tổng quát lên không gian nhiều chiều: Khoảng cách từ một điểm (vector) có toạ độ  tới siêu mặt phẳng (hyperplane) có phương trình  được xác định bởi:

Trong đó: với d là số chiều của không gian.

### 4.1.2. SVM cho 2 lớp dữ liệu

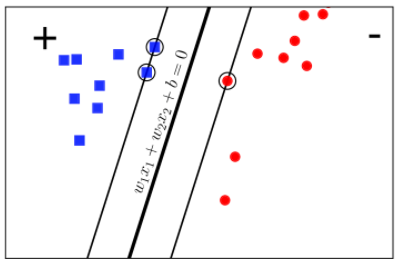
Với 2 lớp dữ liệu, được coi là linearly separable, tức là tồn tại một siêu phẳng phân chia chính xác hai class đó. Việc tìm tìm siêu phẳng phân chia hai class, tức tất cả các điểm thuộc một class nằm về cùng một phía của siêu mặt phẳng đó và ngược phía với toàn bộ các điểm thuộc class còn lại. Chúng ta đã biết rằng, thuật toán PLA có thể làm được việc này nhưng nó có thể cho chúng ta vô số nghiệm như dưới đây:



Hình 1. Các mặt phân cách hai class linearly separable

Dựa vào hình trên ta thấy rằng có rất nhiều mặt phẳng phân chia, vậy đâu là mặt phẳng tối ưu nhất hay làm cho các lớp dữ liệu cách xa mặt phẳng đó nhất. Trước khi tìm được mặt phẳng ta cùng tìm hiểu khái niệm: margin, soft margin, hard margin.

* **Margin:** khoảng cách giữa siêu phẳng (trong trường hợp không gian 2 chiều là đường thẳng) đến 2 điểm dữ liệu gần nhất tương ứng với 2 lớp.



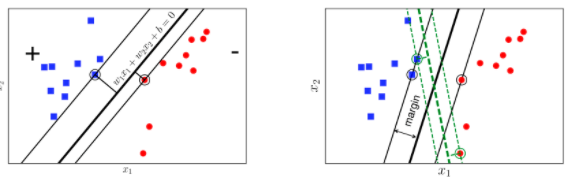
SVM cố gắng tối ưu thuật toán bằng [2] các tìm cách maximize giá trị margin này, từ đó tìm ra siêu phẳng **đẹp nhất** để phân 2 lớp dữ liệu.

Bài toán tìm Margin cực đại là một **Quadratic Programming**, được giải bằng cách giải bài toán đối ngẫu Lagrange (Lagrange dual problem).’

Cách tính Margin:

* Trong không gian 2 chiều:
* Trong không gian nhiều chiều:

Xét hai hình sau:

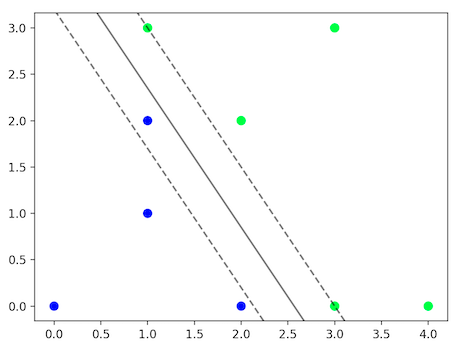


Hình 2. Margin của hai class là bằng nhau và lớn nhất có thể

Nếu ta định nghĩa mức độ hạnh phúc của một class tỉ lệ thuận với khoảng cách gần nhất từ một điểm của class đó tới đường/mặt phân chia thì ở hình bên trái, class tròn đỏ sẽ không được hạnh phúc cho lắm vì đường phân chia gần nó hơn class vuông xanh rất nhiều. Chúng ta cần một đường phân chia sao cho khoảng cách từ điểm gần nhất của mỗi class (các điểm được khoanh tròn) tới đường phân chia là như nhau, như thế thì mới công bằng. Khoảng cách như nhau này được gọi là margin (lề).

Chúng ta xét tiếp hình bên phải khi khoảng cách từ đường phân chia tới các điểm gần nhất của mỗi class là như nhau. Xét hai cách phân chia bởi đường nét liền màu đen và đường nét đứt màu lục, đường nào sẽ làm cho cả hai class hạnh phúc hơn? Rõ ràng đó phải là đường nét liền màu đen vì nó tạo ra một margin rộng hơn.

* **Soft margin và hard margin:** để tránh **overfitting**, nhiều khi để muốn có margin cao, ta chấp nhận việc một vài data có thể không được chia chính xác (ví dụ như 1 bóng xanh bị lọt sang vùng của bóng đỏ). Data này được gọi là nhiễu.



Hình 3. Soft margin

Margin trong trường hợp này gọi là **Soft Margin**.  
**Hard Margin** ám chỉ việc tìm được Margin mà không nhiễu (tất cả các data đều thoả mãn sự phân chia). Với các bái toán thực tế, việc tìm được **Hard Margin** nhiều khi là bất khả thi, vì thế việc chấp nhận sai lệch ở một mức độ chấp nhận được là vô cùng cần thiết.

Trong cài đặt SVM, người ta giới thiệu tham số CC với quy ước:

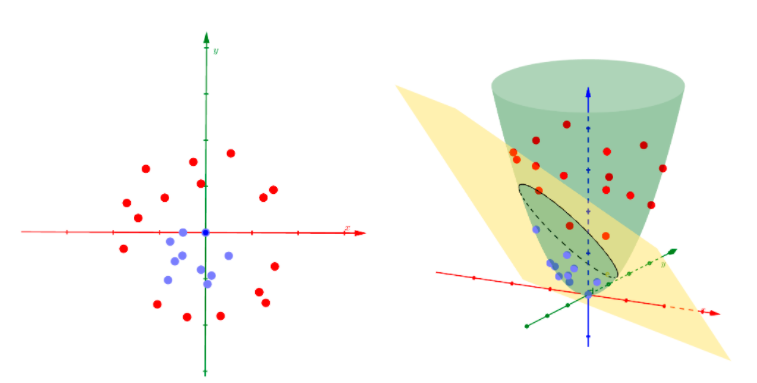
* **C = ∞**: không cho phép sai lệch, đồng nghĩa với **Hard Margin**.
* **C lớn**: cho phép sai lệch nhỏ, thu được Margin nhỏ.
* **C nhỏ**: cho phép sai lệch lớn, thu được Margin lớn.

Tuỳ bài toán cụ thể mà ta cần **điểu chỉnh tham số C này để thu được kết quả tốt nhất**.

Việc margin **rộng hơn sẽ mang lại hiệu ứng phân lớp tốt hơn** vì sự phân chia giữa hai class là rạch ròi hơn. Nhưng trên thực tế bộ dữ liệu dữ liệu của chúng ta phức tạp hơn rất nhiều.

### 4.1.3. Kernel SVM

Phần trên nhóm tôi đã tìm hiểu SVM trong việc phân chia cho 2 lớp dữ liệu, nhưng trong thực tế lượng dữ liệu của ta lớn hơn rất nhiều vì vậy việc phân chia là vô cùng khó khăn. Nói cách khác dữ liệu là dữ liệu không tuyến tính.



Hình 4. Dữ liệu không tuyến tính được ánh xạ sang một không gian khác

Hình bên trái dữ liệu của hai classes là không phân biệt tuyến tính trong không gian hai chiều.

Hình bên phải nếu coi thêm chiều thứ ba là một hàm số của hai chiều còn lại z=x2+y2, các điểm dữ liệu sẽ được phân bố trên 1 parabolic và đã trở nên phân biệt tuyến tính. Mặt phẳng màu vàng là mặt phân chia, có thể tìm được bởi Hard/Soft Margin SVM.

Nói một cách ngắn gọn, Kernel SVM là việc đi tìm một hàm số biến đổi dữ liệu xx từ không gian feature ban đầu thành dữ liệu trong một không gian mới bằng hàm số Φ(x). Trong ví dụ này, hàm Φ() Φ() đơn giản là giới thiệu thêm một chiều dữ liệu mới (một feature mới) là một hàm số của các features đã biết. Hàm số này cần thỏa mãn mục đích của chúng ta: trong không gian mới, dữ liệu giữa hai classes là phân biệt tuyến tính hoặc gần như phần biệt tuyến tính. Khi đó, ta có thể dùng các bộ phân lớp tuyến tính thông thường như PLA, Logistic Regression, hay Hard/Soft Margin SVM.

Nếu phải so sánh, ta có thể thấy rằng hàm biến đổi Φ() tương tự như activation functions trong Neural Networks. Tuy nhiên, có một điểm khác biệt ở đây là: trong khi nhiệm vụ của activation function là phá vỡ tính tuyến tính của mô hình, hàm biến đổi Φ() đi biến dữ liệu không phân biệt tuyến tính thành phân biệt tuyến tính. Như vậy là để đạt được mục đích chung, ta có hai cách nhìn khác nhau về cách giải quyết.

Các hàm Φ() thường tạo ra dữ liệu mới có số chiều cao hơn số chiều của dữ liệu ban đầu, thậm chí là vô hạn chiều. Nếu tính toán các hàm này trực tiếp, chắc chắn chúng ta sẽ gặp các vấn đề về bộ nhớ và hiệu năng tính toán. Có một cách tiếp cận là sử dụng các kernel functions mô tả quan hệ giữa hai điểm dữ liệu bất kỳ trong không gian mới, thay vì đi tính toán trực tiếp từng điểm dữ liệu trong không gian mới. Kỹ thuật này được xây dựng dựa trên quan sát về bài toán đối ngẫu của SVM.

**Cơ sở toán học:**

λ=arg subject to:

Trong đó:

* N: số cặp điểm dữ liệu trong tập training.
* xn: feature vector của dữ liệu thứ nn trong tập training.
* yn: nhãn của dữ liệu thứ n, bằng 1 hoặc -1.
* λn: nhân tử Lagrange ứng với điểm dữ liệu thứ n.
* C: hằng số dương giúp cân đối độ lớn của margin và sự hy sinh của các điểm nằm trong vùng không an toàn. Khi C=∞ hoặc rất lớn, Soft Margin SVM trở thành Hard Margin SVM.

### 4.1.4. Ưu điểm của SVM

Là một kĩ thuật phân lớp khá phổ biến, SVM thể hiện được nhiều ưu điểm trong số đó có việc tính toán hiệu quả trên các tập dữ liệu lớn. Có thể kể thêm một số ưu điểm của phương pháp này như:

* **Xử lý trên không gian số chiều cao**: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn
* **Tiết kiệm bộ nhớ**: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết dịnh
* **Tính linh hoạt** - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

### 4.1.5. Nhược điểm của SVM

* **Bài toán số chiều cao**: Trong trường hợp số lượng thuộc tính (**p**) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (**n**) thì SVM cho kết quả khá tồi
* **Chưa thể hiện rõ tính xác suất**: Việc phân lớp của SVM chỉ là việc cố gắng tách các đối tượng vào hai lớp được phân tách bởi siêu phẳng SVM. Điều này chưa giải thích được xác suất xuất hiện của một thành viên trong một nhóm là như thế nào. Tuy nhiên hiệu quả của việc phân lớp có thể được xác định dựa vào khái niệm **margin** từ điểm dữ liệu mới đến siêu phẳng phân lớp mà chúng ta đã bàn luận ở trên.

**4.2. Tensorflow**

### 4.2.1. Định nghĩa

Tensorflow được hiểu là một thư viện nguồn mở và được sử dụng phổ biến cho việc học máy - Machine Learning trở nên nhanh chóng và dễ dàng hơn. Khi trí tuệ nhân tạo được đưa vào giảng dạy tại các trường Đại học nổi tiếng hàng đầu thế giới và trở nên phổ biến trong giáo dục thì các khái niệm về Tensorflow và các phương pháp học trong trí tuệ nhân tạo cũng vì thế mà trở nên rất phổ biến theo đó.

Tensorflow được tạo ra bởi đội ngũ của Google - Google Brain được tạo ra với mục đích chính đó là sử dụng cho mục đích nghiên cứu và áp dụng trong sản xuất một cách hiệu quả nhất. Tensorflow đã được cấp phép hoạt động vào ngày 9/11/2015.

Tensorflow được sử dụng như một phương tiện trung gian để tính toán các số lượng trong sản xuất và đồng thời là một công cụ không thể thiếu được trong Machine Learning phục vụ cho việc học và nghiên cứu một cách dễ dàng hơn.

### 4.2.2. Các thành phần chính

Để có thể tạo ra một chương trình Tensorflow đơn giản cần phân chia thành 2 phần chính đó là: Construction Phase - được hiểu là một mô hình tính toán và phần thứ hai đó là Execution Phase - được hiểu là chạy mô hình vừa được xây dựng xong. Trong đó các bước thực hiện một chương trình Tensorflow bao gồm các bước thực hiện như sau:

* Bước 1: Xuất Tensorflow bằng thuật toán -> Import Tensorflow as tf. Thực tế cho thấy Tensorflow sử dụng một ngôn ngữ lập trình Python để phục vụ cho việc lập trình nhưng không có sẵn trong đó vì vậy khi sử dụng ngôn ngữ lập trình cần phải xuất Tensorflow.
* Bước 2: Thực hiện xây dựng phương trình toán học để tính hàm F. Phương trình bao gồm các thông tin như sau: X và y được gọi là các nút mạng hay biến số, 2 là hằng số không thay đổi. Dựa vào các quy định về việc xây dựng một phương trình trên bạn có thể thực hiện 3 phương trình toán học bằng các cách khác nhau.
* Bước 3: Chạy mô hình mà bạn vừa xây dựng bằng cách phải tạo thêm được một một hàm tính toán nữa để có thể lập giá trị cho các biến số X và Y để tìm ra đáp án cho hàm F. Tiếp theo, việc bạn cần làm đó chính là sử dụng hàm Global variable Initializer để có thể khởi tạo cho các biến trong thuật toán đó.

### 4.2.3. Ưu điểm

* TensorFlow có cấu trúc đáp ứng khi bạn có thể dễ dàng hình dung từng phần của biểu đồ.
* Nó có tính linh hoạt nền tảng, có nghĩa là nó là mô-đun và một số phần của nó có thể độc lập trong khi các phần khác kết hợp lại.
* Nó có thể dễ dàng huấn luyện trên CPU cũng như GPU để tính toán phân tán.
* Nó có khả năng phân biệt tự động, có lợi cho các thuật toán học máy dựa trên độ dốc, nghĩa là bạn có thể tính toán các đạo hàm của các giá trị đối với các giá trị khác, dẫn đến mở rộng biểu đồ.
* Nó có hỗ trợ nâng cao cho các luồng, tính toán không đồng bộ và hàng đợi.
* Nó là tùy biến và nguồn mở.

## 4.3. Mạng Nơ-ron (Neural Network)

### 4.3.1. Định nghĩa

Mạng nơ-ron nhân tạo là một chuỗi các thuật toán được đưa ra để nỗ lực tìm kiếm các mối quan hệ cơ bản trong một tập hợp dữ liệu, thông qua quá trình bắt chước cách thức hoạt động của bộ não con người.

Theo nghĩa này, mạng nơ-ron nhân tạo được xem như hệ thống các tế bào thần kinh nhân tạo, có thể là hữu cơ hoặc nhân tạo về bản chất.

Mạng nơ-ron nhân tạo có thể thích ứng với các thay đổi trong đầu vào, do đó, nó đưa ra các kết quả tốt nhất có thể mà không cần phải thiết kế lại các tiêu chí đầu ra.

Khái niệm mạng nơ-ron nhân tạo có nguồn gốc từ trí tuệ nhân tạo, đang nhanh chóng trở nên phổ biến trong sự phát triển của các hệ thống giao dịch điện tử.

### 4.3.2. Các đặc điểm của mạng nơ-ron

Mạng nơ-ron nhân tạo trong tài chính, hỗ trợ phát triển các qui trình như dự báo chuỗi thời gian, các giao dịch thuật toán, phân loại chứng khoán, các mô hình rủi ro tín dụng và xây dựng các chỉ báo độc quyền và các công cụ phái sinh giá cả.

Mạng nơ-ron nhân tạo hoạt động tương tự như mạng nơ-ron của con người. Một "nơ-ron thần kinh" trong mạng nơ-ron nhân tạo là một hàm toán học có chức năng thu thập và phân loại thông tin theo một cấu trúc cụ thể.

Mạng nơ-ron nhân tạo có sự tương đồng chuẩn mạnh với các phương pháp thống kê như các đồ thị đường cong và phân tích hồi qui.

Mạng nơ-ron nhân tạo chứa các lớp bao hàm các nút (node) được liên kết với nhau. Mỗi nút là một tri giác (hay một nơ-ron nhân tạo), cấu tạo tương tự như một hàm hồi qui đa tuyến tính.

Tri giác sẽ cung cấp tín hiệu được tạo bởi hàm hồi qui đa tuyến tính, tạo thành một hàm kích hoạt (có thể là phi tuyến).

Trong một tri giác đa lớp (MLP), các tri giác sẽ được sắp xếp theo các lớp liên kết với nhau. Lớp đầu vào thu thập các mẫu đầu vào, và lớp đầu ra nhận các phân loại hoặc tín hiệu đầu ra mà các mẫu đầu vào có thể phản ánh.

Ví dụ, mô hình có thể có đầu vào là một danh sách các đại lượng chỉ báo kĩ thuật về một chứng khoán nhất định, kết quả đầu ra tiềm năng có thể là các đề xuất "Mua", "Giữ" hoặc "Bán".

### 4.3.3. Ứng dụng của mạng Nơ-ron

Mạng nơ-ron nhân tạo được sử dụng rộng rãi ở nhiều lĩnh vực, có thể ứng dụng cho tài chính, lập kế hoạch doanh nghiệp, giao dịch, phân tích kinh doanh và bảo trì sản phẩm.

Mạng nơ-ron nhân tạo cũng được sử dụng rộng rãi trong các hoạt động kinh doanh như dự báo và tìm kiếm giải pháp nghiên cứu tiếp thị, phát hiện gian lận và đánh giá rủi ro.

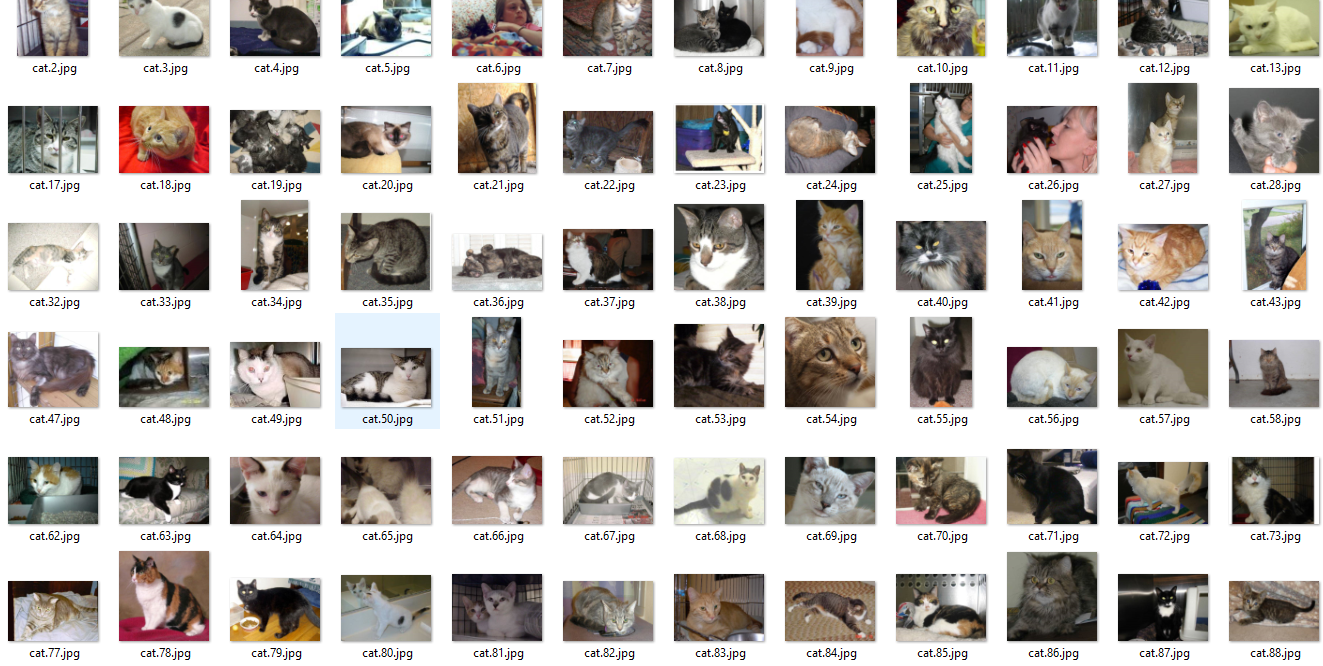
Mạng nơ-ron nhân tạo có thể đánh giá dữ liệu giá và khai quật các cơ hội giao dịch dựa trên phân tích dữ liệu lịch sử.

Mạng nơ-ron nhân tạo cũng có thể phân biệt sự phụ thuộc phi tuyến lẫn nhau của đầu vào mà các mô hình phân tích kĩ thuật khác không thể làm được.

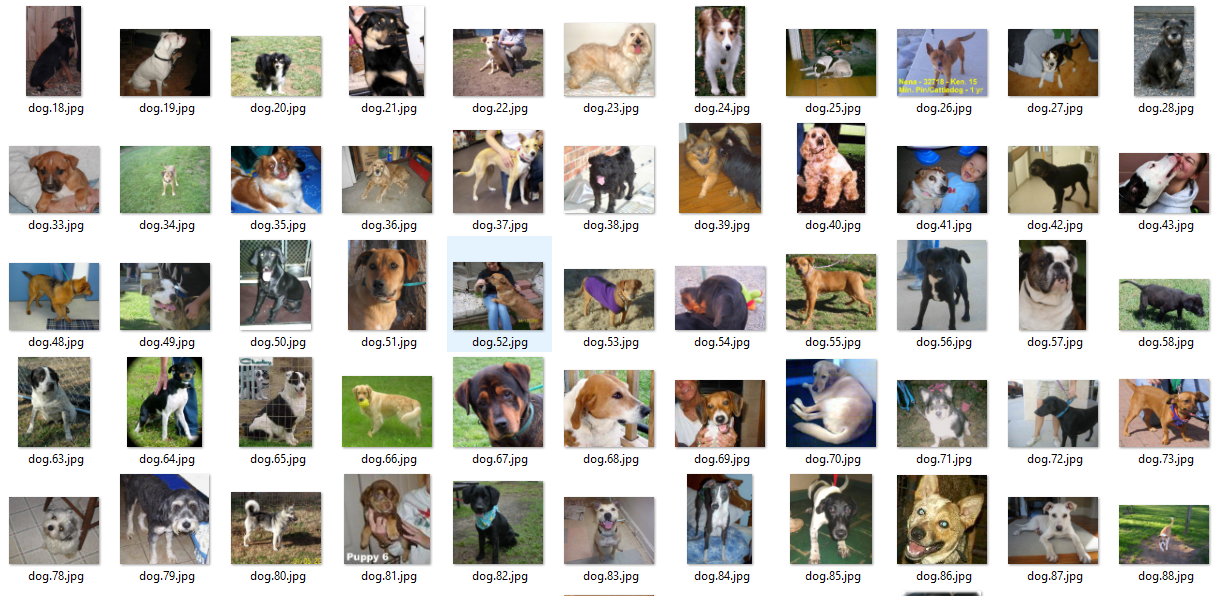
Tuy nhiên, tính chính xác trong việc sử dụng mạng nơ-ron nhân tạo để đưa ra dự đoán giá cho cổ phiếu là khác nhau. Một số mô hình dự đoán giá cổ phiếu chính xác 50 đến 60% thời gian, trong khi những mô hình khác có tính chính xác lên đến 70% trong tất cả các trường hợp.

## 4.4. Cài đặt chương trình

### 4.4.1. Dataset của chó và mèo



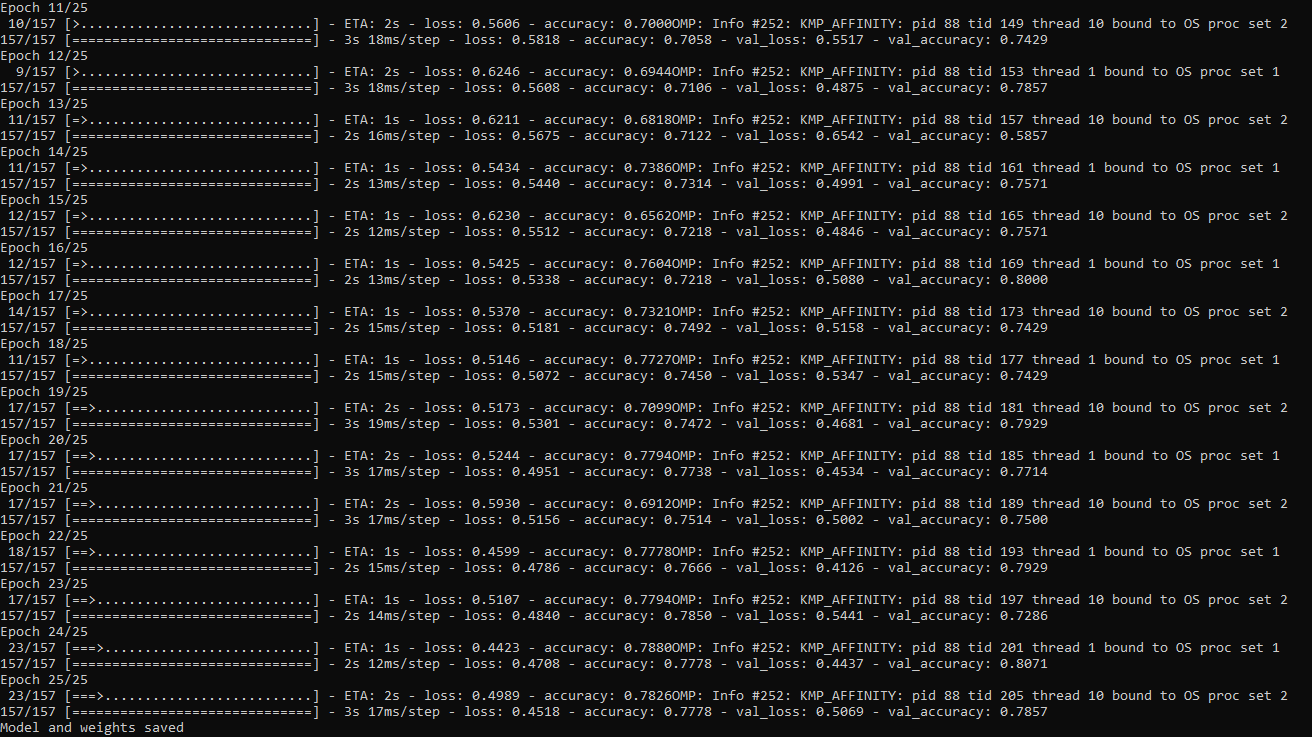
Hình 5. Dataset mèo



Hình 6. Dataset chó

### 4.4.2. Train Data

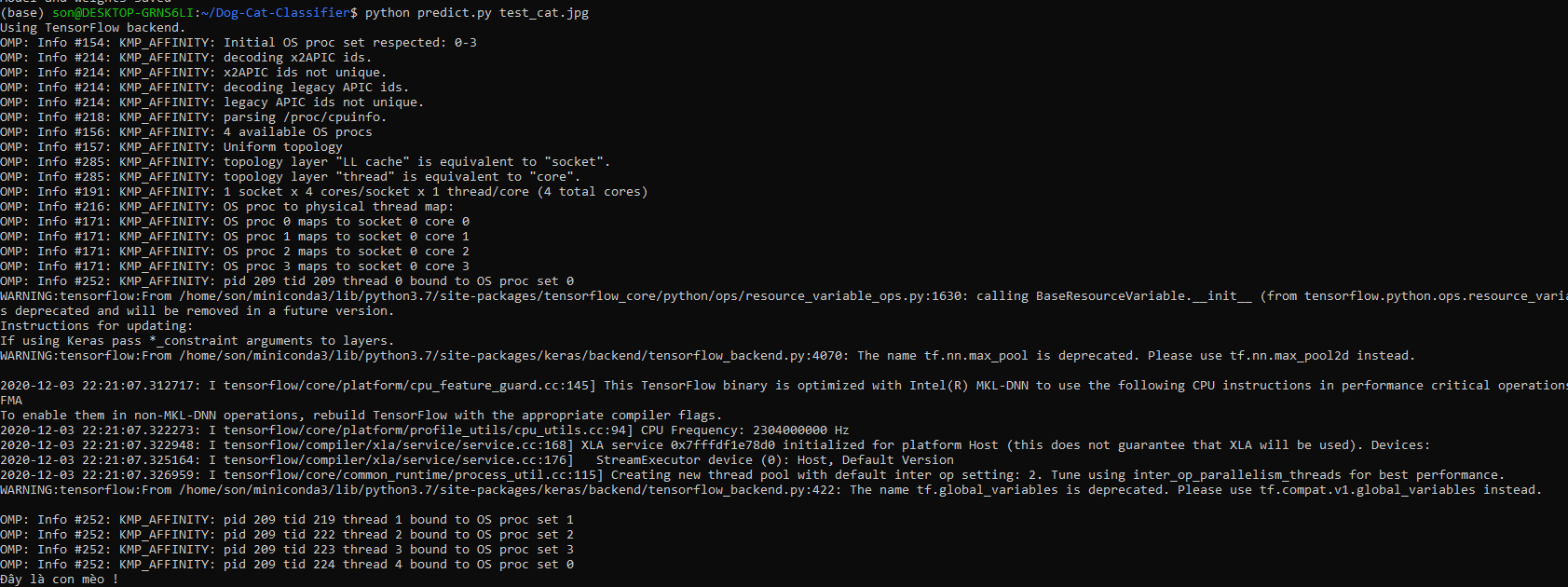
Chạy câu lệnh trên CMD: python train.py



Hình 7. Train Data

### 4.4.2. Predict

Chạy câu lệnh trên CMD: python predict.py <tên file ảnh chó hoặc mèo>

**

Hình 8. Predict Data

# LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn chân thành đến thầy **Trần Chí Kiên** đã giúp đỡ bọn em rất nhiều trong việc học cũng như bài tập lớn này để bọn em có thể hoàn thành nó đúng thời hạn. Chúng em đã nhận được sự quan tâm giúp đỡ hướng dẫn rất tận tình của thầy. Do chưa có nhiều kinh nghiệm làm để tài cũng như những hạn chế về kiến thức, bài làm chắc chắn sẽ không tránh khỏi những thiếu sót. Rất mong nhận được sự nhận xét, ý kiến đóng góp, phê bình từ phía Thầy để bài tập của chúng em được hoàn thiện hơn.

Lời cuối cùng, em xin kính chúc thầy nhiều sức khỏe, thành công và hạnh phúc.

**TÀI LIỆU THAM KHẢO**

1. P.A. Devijver and J. Kittler, Pattern Recognition;
2. T. Mitchell, Machine Learning,McGraw-Hill 1997;
3. B. Ripley, Pattern Recognition and Neural Networks, Cambridge U.Press 1996;
4. C. Bishop,Neural Networks for Pattern Recognition, Oxford University Press 1995;
5. R.O. Duda and P.E. Hart, Pattern Classification and Scene Analysis, 1973
6. V. Vapnik, Ya Chervonenkis, Pattern Recognition, Nauka 1974
7. V. Vapnik, Estimation of Dependencies Based on Empirical Data, Springer Verlag 1982;
8. Statistical Pattern Recognition,Jain, Duin and Mao, IEEE Tran. Patt. Anal. and Mach. Intellig., 22:4-37, 2000