|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  **KHOA CÔNG NGHỆ THÔNG TIN**    Giới Thiệu Trường Đại Học Giao Thông Vận Tải - UTC  **BÁO CÁO**  **THỰC TẬP TỐT NGHIỆP**   |  |  | | --- | --- | | **Giảng viên hướng dẫn : TS. Nguyễn Quốc Tuấn** |  | | **Sinh viên thực hiện : Nguyễn Văn Nghĩa** |  | | **Lớp : CNTT3 – K62** |  | | **Mã sinh viên : 211213161** |  |   **Hà Nội – 2025** |

**MỤC LỤC**

[LỜI MỞ ĐẦU 3](#_Toc190295031)

[CHƯƠNG I: TỔNG QUAN VỀ HỌC MÁY 4](#_Toc190295032)

[I. Lịch sử phát triển 4](#_Toc190295033)

[II. Định nghĩa về học máy 5](#_Toc190295034)

[III. Phân loại học máy 5](#_Toc190295035)

[1. Các loại học máy 5](#_Toc190295036)

[2. Học có giám sát (Supervised Learning) 6](#_Toc190295037)

[3. Học không giám sát (Unsupervised Learning) 7](#_Toc190295038)

[4. Học bán giám sát (Semi-supervised Learning) 9](#_Toc190295039)

[5. Học tăng cường (Reinforcement Learning) 10](#_Toc190295040)

[IV. Các bước triển khai một bài toán học máy 11](#_Toc190295041)

[V. Ứng dụng của học máy trong thực tế 12](#_Toc190295049)

[CHƯƠNG II: CÁC CÔNG NGHỆ VÀ CÔNG CỤ LIÊN QUAN 13](#_Toc190295050)

[I. Ngôn ngữ lập trình 13](#_Toc190295051)

[II. Các framework và thư viện 14](#_Toc190295052)

[III. Các nền tảng học máy dựa trên đám mây 16](#_Toc190295053)

[CHƯƠNG III: THÁCH THỨC VÀ HƯỚNG PHÁT TRIỂN 18](#_Toc190295054)

[I. Thách thức trong học máy 18](#_Toc190295055)

[II. Xu hướng mới trong ngành 18](#_Toc190295056)

[KẾT LUẬN 20](#_Toc190295057)

[TÀI LIỆU THAM KHẢO 21](#_Toc190295058)

# 

# **LỜI MỞ ĐẦU**

Trong thời đại số hóa như hiện nay, nơi mà lượng dữ liệu được tạo ra mỗi ngày là vô cùng lớn thì việc quản lý và xử dụng nguồn dữ liệu ấy là một nhu cầu cấp thiết của mỗi doanh nghiệp, cơ quan hay các tổ chức. Và để tận dụng được những giá trị to lớn chứa trong lượng dữ liệu khổng lồ kia thì học máy (Machine Learning) chính là “chiếc chìa khóa vạn năng” giúp con người mở ra cánh cửa trong việc khai thác tri thức từ những dữ liệu ấy. Học máy không chỉ giúp chúng ta phân tích thông tin mà còn có khả năng dự đoán xu hướng, tự động hóa quy trình và tối ưu hóa các quyết định kinh doanh. Nhờ có học máy, ngành y tế có thể chẩn đoán bệnh sớm hơn, các công ty thương mại điện tử có thể cá nhân hóa trải nghiệm mua sắm, và các nền tảng mạng xã hội có thể gợi ý nội dung phù hợp cho từng người dùng. Không chỉ dừng lại ở các ứng dụng thương mại, học máy còn đóng vai trò quan trọng trong nghiên cứu khoa học, giúp con người khám phá những điều chưa từng biết đến. Với tốc độ phát triển hiện nay, học máy không chỉ là một xu hướng mà còn là một công cụ thay đổi cách thế giới vận hành.

# **CHƯƠNG I: TỔNG QUAN VỀ HỌC MÁY**

## Lịch sử phát triển

* + - 1. Những ngày đầu của học máy (1943- 1950).
* Năm 1943, Warren McCulloch và Walter Pitts giới thiệu mô hình toán học đầu tiên về mạng nơ-ron nhân tạo.
* Năm 1949, Donald Hebb xuất bản cuốn sách “The Organization of Behavior”, đặt nền tảng cho học máy.
* Năm 1950, Alan Turing đề xuất “Turing Test” trong bài báo “Computing Machinery and Intelligence”, để xác định trí thông minh của máy tính.
  + - 1. Chơi trò chơi và phát triển thuật toán (1952-1981)
* Năm 1952, Arthur Samuel phát triển chương trình học máy đầu tiên: đó là trò chơi cờ đam. Lần đầu tiên thuật ngữ Machine Learning xuất hiện.
* Năm 1957, Frank Rosenblatt phát minh “Perceptron” - mô hình mạng nơ-ron đầu tiên cho máy tính, đặt nền móng cho mạng nơ-ron nhân tạo.
* Năm 1967, thuật toán k-NN (K-Nearest Neighbors) ra đời, một trong những phương pháp phân loại đầu tiên hỗ trợ nhận dạng mẫu và lập biểu đồ lộ trình.
* Vào năm 1979, xe đẩy Stanford - phương tiện tự hành đầu tiên được phát triển tại Đại học Stanford.
* Năm 1981, Gerald Dejong giới thiệu khái niệm “Explanation Based Learning” (EBL) - “ Học tập dựa trên giải thích”, trong đó máy tính phân tích dữ liệu đào tạo và tạo ra một quy tắc chung mà nó có thể tuân theo bằng cách loại bỏ dữ liệu không quan trọng.
  + - 1. Những bước tiến lớn (1990 – 2014).
* Những năm 1990, công việc về máy học đã chuyển từ cách tiếp cận dựa trên kiến ​​thức sang cách tiếp cận dựa trên dữ liệu.
* Năm 1997, Deep Blue của IBM đã gây chấn động thế giới khi đánh bại nhà vô địch cờ vua thế giới.
* Năm 2006, Geoffrey Hinton giới thiệu thuật ngữ “Deep Learning” (Học sâu).
* Năm 2010, Microsoft ra mắt “Kinect”, theo dõi cử động cơ thể theo thời gian thực.
* Năm 2011, IBM Watson đánh bại con người trong “Jeopardy!”. Google Brain được phát triển, học cách nhận diện mèo
* Năm 2012, Google X Lab tạo thuật toán học máy có thể phát hiện mèo trên YouTube.
* Năm 2014, Facebook phát triển DeepFace – thuật toán nhận diện khuôn mặt tương đương con người.
  + - 1. Giai đoạn hiện đại (2015 - nay).
* Năm 2015, Amazon và Microsoft ra mắt nền tảng máy học của riêng mình.
* Năm 2016, AlphaGo được Google DeepMind phát triển đã đánh bại một kỳ thủ cờ vây chuyên nghiệp.
* Năm 2017, Waymo thử nghiệm taxi tự lái đầu tiên tại Phoenix, Mỹ.
* Năm 2020, OpenAI ra mắt GPT-3, mô hình xử lý ngôn ngữ mạnh nhất thế giới với 175 tỷ tham số.
* Năm 2021, Học sâu và học tăng cường phát triển mạnh, AI ứng dụng rộng rãi trong y tế.
* Năm 2022, phát triển mô hình ngôn ngữ đa ngôn ngữ và ứng dụng học máy trong tài chính: phát hiện gian lận, dự đoán xu hướng thị trường…
* Năm 2023, học máy lượng tử đạt tiến bộ, AI tối ưu hóa năng lượng tái tạo.
* Năm 2024, phát triển học máy tự giám sát, AI cá nhân hóa giáo dục.

## Định nghĩa về học máy

* + - 1. Định nghĩa về học máy
* Học máy (Machine Learning) là một nhánh của trí tuệ nhân tạo (AI) tập trung vào việc phát triển các thuật toán giúp máy tính có thể học từ dữ liệu mà không cần lập trình rõ ràng.
  + - 1. Định nghĩa của Arthur Samuel
* Năm 1959, nhà khoa học máy tính Arthur Samuel, người tiên phong trong lĩnh vực học máy, đã định nghĩa về học máy như sau:

“*Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed*”.

*(Học máy là lĩnh vực nghiên cứu giúp máy tính có khả năng học hỏi mà không cần lập trình rõ ràng.)*

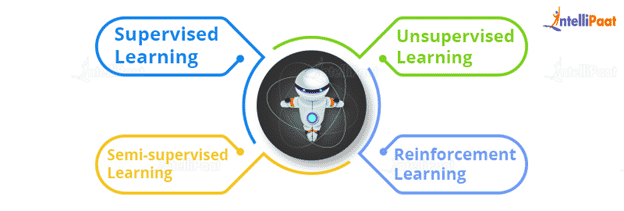
* + - 1. Định nghĩa của Tom Mitchell
* Theo Tom Mitchell (1997), một định nghĩa hiện đại và chặt chẽ hơn về học máy là:

*“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.”*

*(Một chương trình máy tính được cho là học từ kinh nghiệm E với nhiệm vụ T và hiệu suất P, nếu hiệu suất của nó trong nhiệm vụ T, được đo bằng P, cải thiện nhờ kinh nghiệm E.)*

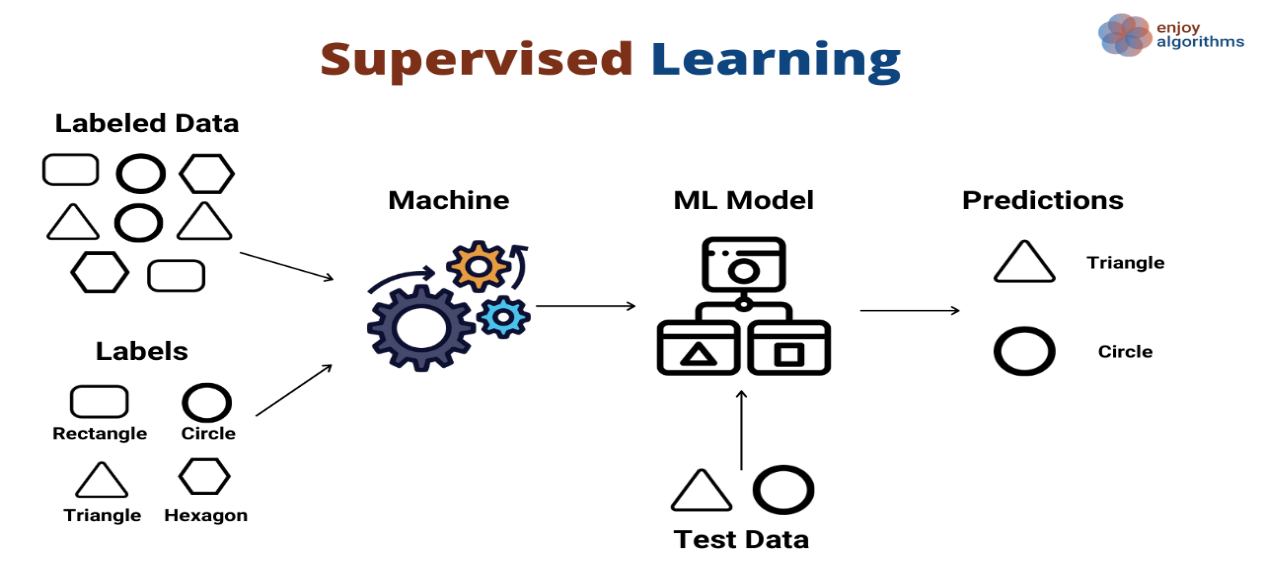
## Phân loại học máy

* + - 1. Các loại học máy
* Học máy gồm có 4 loại phổ biến:
  + Học có giám sát (Supervised Learning)
  + Học không giám sát (Unsupervised Learning)
  + Học bán giám sát (Semi-Supervised Learning)
  + Học tăng cường (Reinforcement Learning)



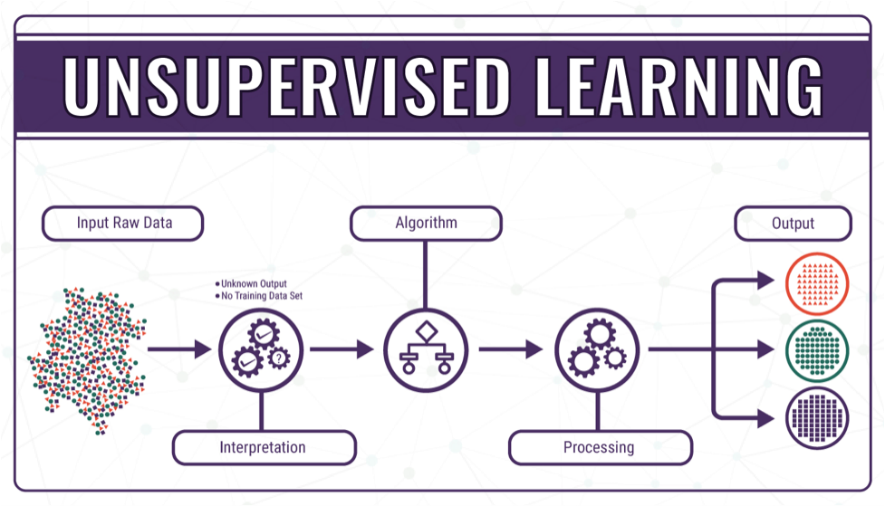
Hình 1.1: Các loại học máy

* + - 1. Học có giám sát (Supervised Learning)



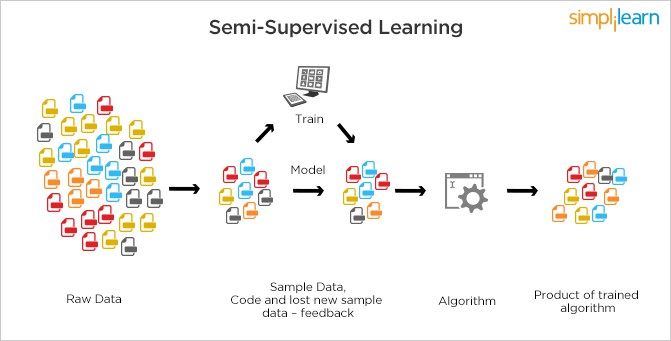
Hình 1.2: Học máy có giám sát (Supervised Learning)

* Đặc điểm: Mô hình học từ dữ liệu đã được gán nhãn (dữ liệu đầu vào) để dự đoán nhãn của dữ liệu mới (dữ liệu đầu ra).
* Mục tiêu: Dự đoán hoặc phân loại giá trị đầu ra dựa trên các dữ liệu đầu vào đã được gán nhãn trước.
* Học có giám sát có 2 nhóm chính:
  + Phân loại (Classification): Dự đoán nhãn thuộc về một nhóm cụ thể
    - Thuật toán phân loại dùng để giải quyết các vấn đề phân loại, trong đó biến đầu ra là phân loại, ví dụ: “Có” hoặc “Không” ; “Nam” hoặc “Nữ”…
    - Thuật toán phân loại dự đoán các danh mục có trong tập dữ liệu.
    - Ứng dụng: Phát hiện thư rác, lọc email, phân loại khách hàng, chẩn đoán bệnh trong y tế…
    - Một số thuật toán phân loại phổ biến: Random Forest, Decision Tree, Logistic Regression, Support Vector Machine (SVM)
  + Hồi quy (Regression): Dự đoán giá trị liên tục
    - Thuật toán hồi quy dùng để giải quyết các vấn đề hồi quy trong đó có mối quan hệ tuyến tính giữa biến đầu vào và đầu ra.
    - Thuật toán hồi quy dùng để dự đoán các biến đầu ra liên tục.
    - Ứng dụng: Dự đoán giá nhà, dự báo thời tiết, dự đoán giá cổ phiếu…
    - Các thuật toán hồi quy phổ biến: Linear Regression, Decision Tree Regression, Lasso Regression, Gradient Boosting Regression…
* Ưu điểm và thách thức của học có giám sát:
  + Ưu điểm:
    - Độ chính xác cao nếu dữ liệu được huấn luyện tốt.
    - Dữ liệu huấn luyện được gán nhãn rõ ràng, thuận tiện cho việc đánh giá.
    - Ứng dụng cho nhiều bài toán thực tế: dự đoán giá nhà, phân loại email, nhận diện giọng nói…
    - Dễ dàng tối ưu hóa mô hình bằng cách thu thập thêm dữ liệu hoặc tinh chỉnh tham số mô hình.
  + Thách thức:
    - Có thể bị quá khớp (overfitting) nếu dữ liệu không đa dạng.
    - Tốn thời gian và công sức gán nhãn cho dữ liệu.
    - Kém linh hoạt với dữ liệu mới hoặc không được gán nhãn.
    - Cần tài nguyên tính toán lớn cho các thuật toán phức tạp.
* Ứng dụng thực tế của học có giám sát:
  + Phân đoạn hình ảnh
  + Chẩn đoán y khoa
  + Phát hiện gian lận
  + Phát hiện thư rác (email spam)
  + Nhận diện giọng nói
    - 1. Học không giám sát (Unsupervised Learning)



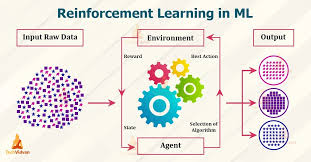
Hình 1.3: Học máy không giám sát (Unsupervised Learning)

* Đặc điểm: Mô hình được huấn luyện dựa trên dữ liệu không được gán nhãn.
* Mục tiêu: Tự tìm ra các mẫu, cấu trúc ẩn trong dữ liệu mà không có nhãn.
* Học không giám sát gồm 3 nhóm:
  + Phân cụm (Clustering)
    - Nhóm các điểm dữ liệu có đặc điểm giống nhau thành từng cụm
    - Ứng dụng: Phân nhóm khách hàng, phát hiện gian lận trong giao dịch tài chính, nhận diện văn bản…
    - Các thuật toán phân cụm phổ biến: K-means, DBSCAN, Hierarchical Clustering.
  + Giảm chiều dữ liệu (Dimensionality Reduction)
    - Giảm số lượng biến đầu vào nhưng vẫn giữ lại thông tin quan trọng
    - Ứng dụng: Xử lý ảnh, tăng tốc học máy, trực quan hóa dữ liệu nhiều chiền trên đồ thị 2D, 3D…
    - Các thuật toán giảm chiều dữ liệu phổ biến: PCA, t-SNE, LDA
  + Luật kết hợp (Association Rule)
    - Tìm mối quan hệ giữa các mục trong tập dữ liệu
    - Ứng dụng thực tế: Gợi ý sản phẩm, quản lý kho hàng, phát hiện gian lận…
    - Các thuật toán luật kết hợp phổ biến: Apriori, FP-Growth.
* Ưu điểm và thách thức của học không giám sát:
  + Ưu điểm:
    - Không cần dữ liệu được gán nhãn thủ công
    - Có khả năng tìm ra các mẫu ẩn trong tập dữ liệu
    - Phân loại dữ liệu nhiễu và phức tạp
    - Giảm chiều dữ liệu làm cho việc xử lý và trực quan hóa dữ liệu dễ dàng hơn
  + Thách thức:
    - Khó diễn giải và giải thích
    - Khó đánh giá độ chính xác và hiệu suất của mô hình
    - Yêu cầu tài nguyên tính toán
* Ứng dụng thực tế:
  + Hệ thống gợi ý sản phẩm
  + Phát hiện gian lận trong giao dịch
  + Phân khúc khách hàng trong kinh doanh
  + Nhận diện khuôn mặt
    - 1. Học bán giám sát (Semi-supervised Learning)



Hình 1.4: Học máy bán giám sát (Semi-supervised Learning)

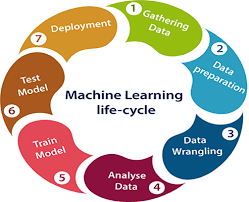
* Đặc điểm:
  + Là nền tảng trung gian giữa học có giám sát và học không giám sát
  + Kết hợp dữ liệu được gán nhãn và không được gán nhãn trong suốt quá trình huấn luyện.
* Mục tiêu: sử dụng hiệu quả tất cả dữ liệu có sẵn, thay vì chỉ dữ liệu được gắn nhãn như trong học có giám sát.
* Ưu điểm và thách thức của học bán giám sát:
  + Ưu điểm:
    - Giảm chi phí và công sức gán nhãn dữ liệu.
    - Tận dụng hiệu quả dữ liệu không nhãn
    - Tăng hiệu suất mô hình
  + Thách thức:
    - Khó triển khai trong môi trường thực tế
    - Phụ thuộc vào giả định về mối quan hệ giữa dữ liệu có nhãn và không nhãn
    - Khó điều chỉnh nếu dữ liệu không đồng nhất:
* Các mô hình tiêu biểu:
  + Self-training
  + Consistency Regularization
  + Graph-based Learning
  + Generative Models (GANs, Variational Autoencoders)
* Ứng dụng:
  + Nhận diện hình ảnh
  + Nhận diện giọng nói
  + Phân loại văn bản
    - 1. Học tăng cường (Reinforcement Learning)



Hình 1.5: Học máy tăng cường (Reinforcement Learning)

* Đặc điểm:
  + Học tăng cường hoạt động dựa trên phản hồi, nơi một tác nhân (agent) tự học bằng cách thử nghiệm & nhận phản hồi từ môi trường.
  + Tác nhân được thưởng cho mỗi hành động tốt và bị phạt cho mỗi hành động xấu.
* Mục tiêu: Tối đa hóa phần thưởng theo thời gian.
* Ưu điểm và thách thức của học tăng cường:
  + Ưu điểm:
    - Giải quyết bài toán phức tạp
    - Không cần dữ liệu có nhãn
    - Học hỏi và cải thiện liên tục
  + Thách thức:
    - Thời gian huấn luyện dài
    - Phụ thuộc vào hệ thống phần thưởng
    - Khó thích nghi với môi trường biến động
* Các mô hình tiêu biểu:
  + Q-Learning
  + Deep Q-Networks (DQN)
  + Proximal Policy Optimization (PPO)
* Ứng dụng:
  + Chơi game: AlphaGo, Chess AI
  + Xe tự hành
  + Hệ thống đề xuất, gợi ý sản phẩm

## Các bước triển khai một bài toán học máy



Hình 1.6: Các bước trong vòng đời học máy

* + - 1. Thu thập dữ liệu (Gathering Data)
* Xác định yêu cầu và phạm vi bài toán
* Xác định nguồn dữ liệu cần thu thập: từ cơ sở dữ liệu, API, web scraping hoặc bộ dữ liệu có sẵn
* Kiểm tra tính có sẵn, khối lượng và chất lượng của dữ liệu.
  + - 1. Chuẩn bị dữ liệu (Data Preparation)
* Loại bỏ các dữ liệu trùng lặp, dữ liệu không cần thiết.
* Xử lý các giá trị bị thiếu trong tập dữ liệu bằng cách: loại bỏ hoặc điền bằng giá trị trung bình, trung vị, mode hoặc các phương pháp nâng cao.
* Mã hóa các biến phân loại bằng các phương pháp: OneHot Encoding, Label Encoding…
* Chuẩn hóa dữ liệu bằng các phương pháp: Scailing hoặc Normalization nếu cần thiết.
  + - 1. Xử lý dữ liệu (Data Wrangling)
* Loại bỏ các giá trị ngoại lai (outliers) trong tập dữ liệu nếu cần thiết
* Xử lý các dữ liệu mất cân bằng: Oversampling, Undersampling, SMOTE.
* Chia tập dữ liệu thành các tập: tập huấn luyện (train set), tập đánh giá (validation set) và tập kiểm tra (test set) theo tỉ lệ khác nhau: 80-10-10, 70-15-15…
* Chọn và tạo các đặc trưng (Feature Selection & Feature Engineering) để cải thiện hiệu suất mô hình.
  + - 1. Phân tích dữ liệu (Analyse Data)
* Thực hiện phân tích dữ liệu khám phá (EDA) để kiểm tra sự tương quan của các biến và sự phân bố của dữ liệu.
* Vẽ các biểu đồ trực quan hóa như: heatmap, boxplot, histogram, scatterplot để phát hiện xu hướng, khám phá những thông tin quan trọng trong tập dữ liệu.
  + - 1. Lựa chọn và huấn luyện mô hình (Train Model)
* Chọn các mô hình, thuật toán phù hợp với bài toán: Decision Tree, Random Forest, Linear Regression, XGBoost…
* Thực hiện chạy thử nghiệm tập dữ liệu trên nhiều mô hình khác nhau để chọn mô hình tốt nhất.
* Tinh chỉnh siêu tham số (Hyperparameter Tuning) bằng Grid Search, Random Search hoặc Bayesian Optimization.
* Kiểm tra hiệu suất mô hình trên tập validation.
  + - 1. Đánh giá mô hình (Test Model)
* Sử dụng các chỉ số đánh giá phù hợp với bài toán:
  + Phân loại: Accuracy, Precision, Recall, F1-score, ROC-AUC.
  + Hồi quy: RMSE, MAE, R².
* So sánh hiệu suất của các mô hình khác nhau.
* Kiểm tra độ ổn định của mô hình bằng kỹ thuật Cross-Validation hoặc K-Fold.
* Điều chỉnh lại mô hình nếu cần (fine-tuning, thử nghiệm các đặc trưng mới).
  + - 1. Triển khai mô hình (Deployment)
* Đưa mô hình vào môi trường thực tế (web API, ứng dụng, dashboard, v.v.).
* Giám sát hiệu suất mô hình theo thời gian.
* Cập nhật mô hình khi dữ liệu mới xuất hiện hoặc hiệu suất giảm sút.

## Ứng dụng của học máy trong thực tế

* Phân loại và lọc thư rác (email spam)
* Nhận diện hình ảnh, giọng nói (Image/Speech Recognition)
* Hệ thống đề xuất, gợi ý sản phẩm (Product recommendations)
* Xe ô tô tự lái (Self-driving cars)
* Trợ lí ảo (Virtual Personal Assistant)

# **CHƯƠNG II: CÁC CÔNG NGHỆ VÀ CÔNG CỤ LIÊN QUAN**

## Ngôn ngữ lập trình

* + - 1. Python



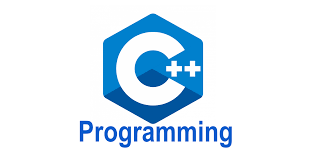
Hình 2.1: Ngôn ngữ lập trình python

* Python là ngôn ngữ lập trình phổ biến nhất trong học máy nhờ cú pháp dễ học, thư viện phong phú.
* Ngôn ngữ này có các thư viện và framework hỗ trợ mạnh mẽ cho học máy và khoa học dữ liệu như: Numpy, Pandas, Matplotlib, Scikit-Learn, TensorFlow, PyTorch…
* Python có cộng đồng người dùng đông đảo, tài liệu phong phú
  + - 1. R



Hình 2.2: Ngôn ngữ lập trình R

* R là ngôn ngữ thích hợp cho phân tích dữ liệu thống kê, trực quan hóa dữ liệu như: caret, randomForest, ggplot2…
* Môi trường tương tác, cho phép thử nghiệm và phát triển ý tưởng nhanh chóng.
* Ngoài ra, R còn có cộng đồng học thuật lớn mạnh.
  + - 1. C++



Hình 2.3: Ngôn ngữ lập trình C++

* Được sử dụng khi cần hiệu suất cao và khả năng kiểm soát phần cứng.
* Tốc độ thực thi cao.
* Khả năng kiểm soát phần cứng.
* Được sử dụng trong các thư viện học máy hiệu quả.
  + - 1. Java

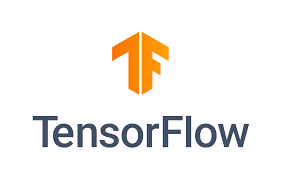


Hình 2.4: Ngôn ngữ lập trình Java

* Java được sử dụng trong các ứng dụng doanh nghiệp.
* Tính di động, có thể chạy trên nhiều nền tảng.
* Khả năng mở rộng, thích hợp cho các dự án lớn.
* Được sử dụng trong các ứng dụng học máy trong doanh nghiệp.

## Các framework và thư viện

* + - 1. TensorFlow



Hình 2.5: Framework TensorFlow

* Được phát triển bởi Google, TensorFlow là một framework mã nguồn mở hàng đầu, cung cấp các công cụ toàn diện cho việc xây dựng và triển khai các ứng dụng học máy.
* Ưu điểm: Tính linh hoạt, khả năng mở rộng, hỗ trợ nhiều nền tảng và cung cấp các API cho nhiều ngôn ngữ lập trình.
* Ứng dụng: Được sử dụng trong nhiều lĩnh vực như nhận dạng hình ảnh, xử lý ngôn ngữ tự nhiên, dịch máy và nhiều ứng dụng khác.
  + - 1. PyTorch



Hình 2.6: Framework PyTorch

* PyTorch là một nền tảng học sâu mã nguồn mở được ưa chuộng, nổi bật với tính linh hoạt và dễ sử dụng.
* Ưu điểm: Giao diện trực quan, khả năng tính toán nhanh chóng trên GPU và tích hợp tốt với các thư viện khác của Python.
* Ứng dụng: Thường được sử dụng trong nghiên cứu và phát triển các mô hình học máy phức tạp.
  + - 1. Scikit-Learn



Hình 2.7: Thư viện Scikit-Learn

* Scikit-learn là một thư viện Python mạnh mẽ, cung cấp một loạt các thuật toán học máy phổ biến và các công cụ hỗ trợ cho việc tiền xử lý dữ liệu, lựa chọn mô hình và đánh giá hiệu suất.
* Ưu điểm: Dễ sử dụng, tài liệu phong phú, phù hợp cho nhiều bài toán học máy khác nhau.
* Ứng dụng: Được sử dụng rộng rãi trong các ứng dụng thực tế như phân loại, hồi quy, gom nhóm và giảm chiều dữ liệu.
  + - 1. Keras



Hình 2.8: Framework Keras

* Keras là một API cấp cao, thân thiện với người dùng để xây dựng và huấn luyện các mạng nơ-ron. Nó có thể chạy trên nhiều nền tảng khác nhau, bao gồm cả TensorFlow và Theano.
* Ưu điểm: Đơn giản, dễ học, cho phép xây dựng mô hình nhanh chóng.
* Ứng dụng: Phù hợp cho người mới bắt đầu và các dự án cần triển khai nhanh.
  + - 1. XGBoost



Hình 2.9: Framework Keras

* XGBoost là một thư viện mạnh mẽ, cung cấp các thuật toán boosting gradient, nổi tiếng với hiệu suất cao và khả năng xử lý dữ liệu lớn.
* Ưu điểm: Độ chính xác cao, tốc độ nhanh, khả năng xử lý dữ liệu thiếu và dữ liệu không cân bằng.
* Ứng dụng: Thường được sử dụng trong các cuộc thi học máy và các bài toán yêu cầu độ chính xác cao.

## Các nền tảng học máy dựa trên đám mây

* + - 1. Google Cloud AI Platform
* Dễ sử dụng, giao diện thân thiện.
* Tích hợp với các dịch vụ khác của Google Cloud Platform (GCP).
* Cung cấp các công cụ AutoML để tự động hóa quá trình học máy.
  + - 1. Amazon SageMaker
* Cung cấp các công cụ và dịch vụ cho quy trình học máy, từ chuẩn bị dữ liệu đến huấn luyện và triển khai mô hình.
* Tích hợp với các dịch vụ khác của Amazon Web Services (AWS).
* Hỗ trợ nhiều framework học máy (TensorFlow, PyTorch, scikit-learn).
  + - 1. Microsoft Azure Machine Learning
* Cung cấp một nền tảng cho học máy, từ xây dựng mô hình đến triển khai và quản lý.
* Tích hợp với các công cụ và dịch vụ của Microsoft.
* Hỗ trợ nhiều framework học máy và ngôn ngữ lập trình.
  + - 1. IBM Watson Studio
* Cung cấp các công cụ cho phân tích dữ liệu và học máy.
* Tích hợp với các dịch vụ khác của IBM Cloud.
* Hỗ trợ nhiều framework học máy và ngôn ngữ lập trình

# 

# **CHƯƠNG III: THÁCH THỨC VÀ HƯỚNG PHÁT TRIỂN**

## Thách thức trong học máy

* + - 1. Chất lượng dữ liệu
* Thiếu dữ liệu: Đôi khi, chúng ta không có đủ dữ liệu để huấn luyện mô hình một cách hiệu quả, đặc biệt là trong các bài toán phức tạp hoặc lĩnh vực mới.
* Dữ liệu không cân bằng: Trong nhiều trường hợp, số lượng mẫu của các lớp khác nhau là không đồng đều, gây khó khăn cho việc huấn luyện mô hình.
* Dữ liệu nhiễu: Dữ liệu huấn luyện có thể chứa các mẫu nhiễu, làm ảnh hưởng đến độ chính xác của mô hình.
* Dữ liệu không đầy đủ: Dữ liệu có thể bị thiếu một số thông tin, gây khó khăn cho việc huấn luyện mô hình.
  + - 1. Mô hình
* Lựa chọn mô hình: Việc chọn mô hình phù hợp cho một bài toán cụ thể là một thách thức.
* Điều chỉnh tham số: Các mô hình học máy thường có nhiều tham số cần được điều chỉnh để đạt hiệu suất tốt nhất.
* Đánh giá mô hình: Việc đánh giá mô hình một cách chính xác là rất quan trọng để đảm bảo mô hình hoạt động tốt trên dữ liệu mới.
  + - 1. Tính toán
* Tốn kém tài nguyên: Huấn luyện các mô hình học máy phức tạp có thể tốn kém tài nguyên tính toán, đặc biệt là khi làm việc với dữ liệu lớn.
* Thời gian huấn luyện: Thời gian huấn luyện mô hình có thể rất lâu, đặc biệt là đối với các mô hình phức tạp và dữ liệu lớn.
  + - 1. Đạo đức và xã hội
* Thiên kiến: Các mô hình học máy có thể bị thiên kiến do dữ liệu huấn luyện bị thiên kiến.
* Quyền riêng tư: Việc sử dụng dữ liệu cá nhân cho học máy cần tuân thủ các quy định về quyền riêng tư.
* Trách nhiệm xã hội: Các mô hình học máy có thể được sử dụng cho các mục đích không tốt, gây ảnh hưởng đến xã hội.

## Xu hướng mới trong ngành

* Học sâu (deep learning): Đây là một lĩnh vực đang phát triển rất nhanh chóng của học máy, với nhiều ứng dụng trong xử lý ảnh, xử lý ngôn ngữ tự nhiên và các lĩnh vực khác.
* Học tăng cường (reinforcement learning): Lĩnh vực này tập trung vào việc huấn luyện các tác tử (agent) để tương tác với môi trường và học hỏi thông qua việc thử và sai.
* Học không giám sát (unsupervised learning): Lĩnh vực này tập trung vào việc tìm kiếm cấu trúc trong dữ liệu không có nhãn.
* Học máy giải thích được (explainable AI): Mục tiêu là làm cho các mô hình học máy dễ hiểu hơn đối với con người.
* Học máy liên tục (continual learning): Lĩnh vực này tập trung vào việc huấn luyện các mô hình có thể học hỏi từ dữ liệu mới mà không quên kiến thức cũ.

# 

# **KẾT LUẬN**

Ngày nay, học máy đang thay đổi cách thức mà con người tương tác với công nghệ và giải quyết vấn đề. Từ những ứng dụng nhỏ trong cuộc sống hàng ngày đến những đột phá khoa học lớn, học máy đã và đang dần trở thành một phần không thể thiếu của xã hội hiện đại. Tuy nhiên, tiềm năng của học máy vẫn còn rất lớn và chưa được khai thác hết. Vì thế vẫn còn nhiều thách thức phía trước như vấn đề đạo đức, bảo mật dữ liệu, và sự tin cậy của các mô hình học máy.

Trong tương lai, chúng ta có thể kỳ vọng vào sự phát triển vượt bậc của học máy, với những ứng dụng ngày càng thông minh và sáng tạo. Học máy sẽ không chỉ giúp chúng ta tự động hóa công việc mà còn mang lại những hiểu biết sâu sắc hơn về thế giới, giúp chúng ta đưa ra những quyết định tốt hơn và giải quyết những vấn đề phức tạp hơn.

# **TÀI LIỆU THAM KHẢO**

<https://www.lightsondata.com/the-history-of-machine-learning/>

<https://www.javatpoint.com/machine-learning-life-cycle>

<https://magnimindacademy.com/blog/machine-learning-tools-and-technologies-1/>

<https://machinelearningmastery.com/7-machine-learning-trends-2025/>

<https://g-customer360.com/thach-thuc-hang-dau-ve-machine-learning/>