ML developing process

Việc phát triển mô hình ML là một **quy trình lặp đi lặp lại** gồm các bước:

1. **Chọn kiến trúc tổng thể**: Gồm việc chọn mô hình, dữ liệu, và siêu tham số.
2. **Huấn luyện mô hình**: Kết quả ban đầu thường chưa đạt như kỳ vọng.
3. **Chẩn đoán mô hình**: Phân tích bias – variance và thực hiện *error analysis* (sẽ nói trong video sau).
4. **Cải tiến mô hình**: Dựa vào chẩn đoán để điều chỉnh kiến trúc, thêm dữ liệu, thay đổi đặc trưng (features), v.v.
5. **Lặp lại quy trình** cho đến khi đạt được hiệu suất mong muốn

Sau huấn luyện, nếu hiệu suất chưa đạt, có nhiều hướng cải tiến:

- **Thu thập thêm dữ liệu**
- **Tạo đặc trưng phức tạp hơn**

Error Analysis

- Là quá trình **xem xét thủ công các ví dụ bị mô hình dự đoán sai** (ví dụ từ tập validation).
- Mục tiêu là **phân nhóm các lỗi theo đặc điểm chung**, từ đó giúp xác định những nguyên nhân chính gây lỗi.

Một số lưu ý trong Error Analysis:

- Các nhóm lỗi **không loại trừ nhau**
- Không cần xem xét toàn bộ ví dụ sai nếu quá nhiều — chỉ cần **lấy mẫu ngẫu nhiên khoảng 100 ví dụ** để có cái nhìn đại diện.

Khi nào Error Analysis hiệu quả?

- Khi bài toán là thứ **con người có thể hiểu rõ và đánh giá được** (như phân loại email).

- Với các bài toán khó với con người (như dự đoán người dùng click vào quảng cáo), phân tích lỗi sẽ **kém hiệu quả hơn**.

Adding data

Thêm dữ liệu có chọn lọc (Targeted Data Collection)

- **Không phải cứ thêm mọi loại dữ liệu đều hiệu quả** – vì có thể rất tốn thời gian và chi phí.
- Dựa vào **error analysis**, chỉ thu thập thêm dữ liệu thuộc các **nhóm lỗi phổ biến** mà mô hình đang gặp khó khăn

Data Augmentation – Tăng dữ liệu bằng biến đổi mẫu hiện có

Data Synthesis – Tạo dữ liệu hoàn toàn mới

- Thay vì biến đổi dữ liệu có sẵn, **tạo ra dữ liệu giả lập hoàn toàn mới**.
- Kỹ thuật này có thể yêu cầu công sức lập trình ban đầu nhưng giúp tạo ra **rất nhiều dữ liệu chất lượng cao**.

Chuyển từ mô hình hướng thuật toán sang mô hình hướng dữ liệu

- Trước đây, hầu hết nghiên cứu học máy là **giữ nguyên dữ liệu, cải tiến thuật toán**.
- Hiện nay, thuật toán đã tốt (logistic regression, neural networks, v.v.), nên **tập trung cải thiện chất lượng và số lượng dữ liệu có thể hiệu quả hơn**.
- Việc **tập trung vào dữ liệu** (data-centric AI) đang trở thành xu hướng quan trọng giúp mô hình học máy hoạt động tốt hơn.

Transfer Learning

Ý tưởng chính

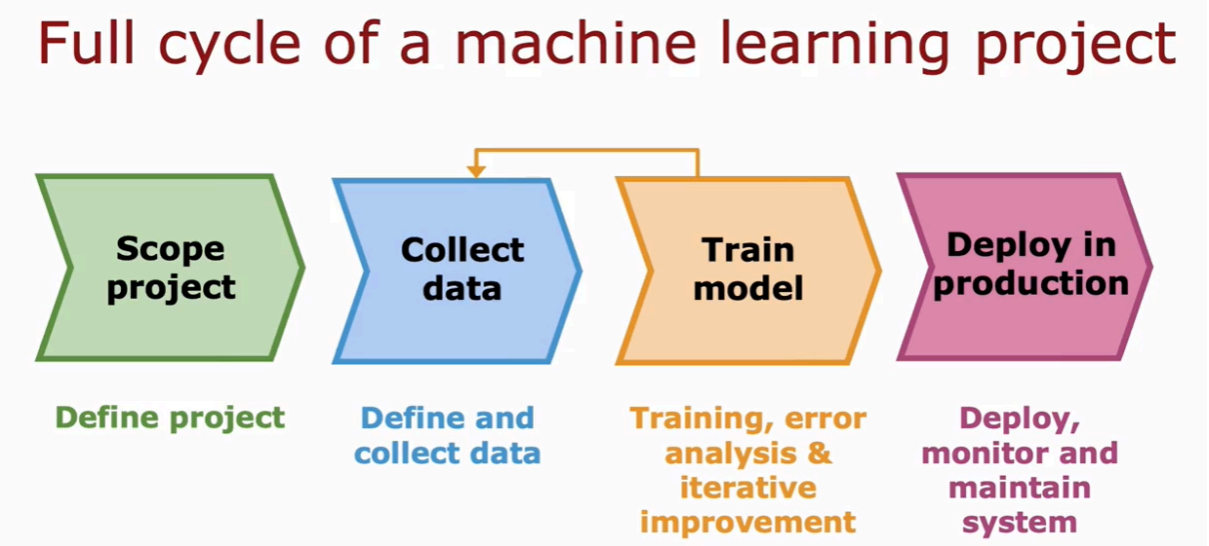
- **Huấn luyện trước (pre-training)** một mô hình trên một tập dữ liệu lớn, ví dụ 1 triệu hình ảnh của mèo, chó, xe, người... với 1000 lớp.
- Sau đó, **sao chép mô hình**, **giữ nguyên các lớp đầu** (layer 1–4) đã học các đặc trưng cơ bản (như cạnh, góc, đường cong).

- **Thay lớp cuối cùng** bằng một lớp mới tương ứng với tác vụ mới (ví dụ: 10 lớp cho các chữ số từ 0 đến 9).
- Tiến hành **tinh chỉnh (fine-tuning)** mô hình trên tập dữ liệu nhỏ của bạn.

Vì sao Transfer Learning hiệu quả?

- Mô hình học từ tập lớn đã học được:
  - Lớp đầu: **phát hiện cạnh (edge)**
  - Lớp giữa: **góc cạnh, hình cong đơn giản**
  - Các lớp sâu hơn: **hình dạng phức tạp**
- Những đặc trưng này **phổ biến ở hầu hết các ảnh**, nên có thể tái sử dụng cho nhiều bài toán thị giác máy khác.

Machine Learning project cycle



1. Scoping – Xác định phạm vi dự án

- Quyết định bạn đang giải quyết bài toán gì.
- Ví dụ: nhận dạng giọng nói cho voice search (tìm kiếm bằng giọng nói).
- Đặt mục tiêu rõ ràng: đầu ra cần là gì? Độ chính xác mong muốn?

2. Data Collection – Thu thập dữ liệu

- Thu thập dữ liệu huấn luyện phù hợp (ví dụ: âm thanh và bản chép lời).

- Xác định nguồn dữ liệu, đảm bảo chất lượng và định dạng đúng.

3. **Model Training – Huấn luyện mô hình**

- Huấn luyện mô hình đầu tiên.
- Thực hiện phân tích lỗi (**error analysis**) hoặc phân tích bias-variance.
- Cải thiện mô hình theo vòng lặp:
  - Phân tích lỗi → Thu thập thêm dữ liệu → Huấn luyện lại

4. **Data Augmentation – Tăng cường dữ liệu (nếu cần)**

- Tạo dữ liệu nhân tạo để mô phỏng môi trường khó khăn hơn (ví dụ: tiếng ồn xe trong giọng nói).

5. **Deployment – Triển khai**

- Triển khai mô hình lên **inference server**.
- Ứng dụng (ví dụ: mobile app) gửi yêu cầu đến server, nhận kết quả dự đoán.
- Cần kỹ năng kỹ thuật phần mềm để đảm bảo server:
  - Có thể xử lý được nhiều người dùng.
  - Tối ưu hóa chi phí tính toán.
  - Ổn định, an toàn và dễ bảo trì.

6. **Monitoring & Maintenance – Giám sát và bảo trì**

- Ghi log dữ liệu vào hệ thống (có sự cho phép từ người dùng).
- Theo dõi hiệu suất của mô hình trong môi trường thực tế.
  - Ví dụ: mô hình không nhận ra tên chính trị gia mới do chưa từng thấy → phải cập nhật lại mô hình.
- **Cập nhật mô hình (Model update)** khi hiệu suất suy giảm hoặc có dữ liệu mới.