# Train AI

## Huấn luyện mô hình

Vòng lặp huấn luyện 5 bước:

Forward pass: Cho mô hình dự báo.

Tính loss: Tính độ sai lệch kết quả dự báo với giá trị thực tế.

Zero gradients: Reset gradient trước mỗi vòng lặp. (Vì gradient bị tích lũy)

Lan truyền ngược (backpropagation): Tính gradient dựa trên hàm loss.

Cập nhật tham số: Dùng optimizer (vd: SGD) để cải thiện trọng số.

Lưu ý (Chọn hàm loss thích hợp)

## Dự báo kết quả

Sử dụng mô hình huấn luyện để dự đoán các giá trị mới trên test set.

Đảm bảo không thay đổi tham số trong quá trình dự báo: sử dụng torch.inference\_mode().

Lưu và tải lại mô hình

Lưu trữ mô hình đã huấn luyện bằng torch.save().

Tải lại bằng torch.load() để tiết kiệm thời gian thay vì huấn luyện lại.

## Mở rộng , Lưu ý

Điều chỉnh learning rate để tối ưu quá trình huấn luyện.

Có thể tích hợp nhiều hàm loss hoặc các optimizers khác nhau (để so sánh hiệu quả).

Đưa thêm regularization (để giảm thiểu overfitting).

Overfitting: Xảy ra khi mô hình học quá kỹ các đặc điểm từ dữ liệu huấn luyện, kể cả nhiễu (noise). Lamf cho mô hình hoạt động rất tốt trên trainning data nhưng kém hiệu quả trên dữ liệu thực tế.

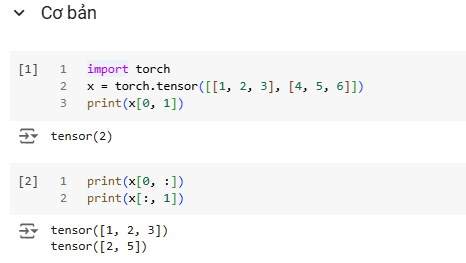
Regularization: Là các kỹ thuật được sử dụng để giảm overfitting

# Indexing

Indexing trong PyTorch là truy cập các phần tensor dựa trên index.

Indexing cơ bản:

Truy cập các phần tử trong tensor qua vị trí cụ thể (giống cách truy cập phần tử trong mảng Numpy).



Indexing bằng Boolean:

Trích xuất phần tử thoả mãng Boolean (True/False).

Ảnh có chứa văn bản, Phông chữ, hàng, màu trắng

Mô tả được tạo tự động

Slicing (Cắt dữ liệu)

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Mô tả được tạo tự động

Gán giá trị

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Mô tả được tạo tự động

# Cross Entropy

## Tổng quan

Cross-Entropy là một hàm loss thường được sử dụng cho các bài toán phân loại. Nó đánh giá mức độ khác biệt giữa 2 phân bố xác xuất. (Trong ML thì được sử dụng làm hàm loss đánh giá sự khác biệt giữa phân phối thực tế và phân phối do model dự đoán)

## Entropy

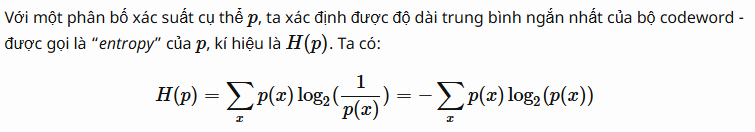
Code: Mã hóa thông tin sang dạng khác ví dụ như nhị phân.

Codeword: Tập hợp kí tự được mã hóa bằng cách thay thế chữ thành chuỗi nhị phân

Encode: Biến đổi dữ liệu gốc thành một dạng mã hóa theo 1 quy tắc

Decode: Giải mã dữ liệu.

Entropy: Cho biết kích cỡ mã hóa trung bình tối thiểu theo một phân phối xác suất, là thước đo sự bất ổn của thông tin. Entropy cao đồng nghĩa với việc ta khó đoán trước

****

p(x): Là phân phối xác suất thực tế

q(x): Là phân phối xác suất mà mô hình dự đoán

## Cross-Entropy

**Cross-Entropy** là một thước đo trong lý thuyết thông tin và học máy để đánh giá sự khác biệt giữa hai phân phối xác suất: **phân phối thực tế** p(x) (ground truth) và **phân phối dự đoán** q(x) (model prediction).

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, hàng

Mô tả được tạo tự động

Tính chất của Cross-Entropy:

- Cross-Entropy luôn lớn hơn Entropy

- Cross-Entropy nhỏ nhất khi p = q

- Không đối xứng

Trong ML:

-Cross Entropy thường được dùng làm hàm loss (nếu q(x) càng giống p(x) thì Cross-Entropy sẽ càng nhỏ và ngược lại do đó nó thường được dùng để làm hàm loss và mục tiêu là tối ưu các tham số sao cho Cross-Entropy là nhỏ nhất).

Input:

* P(x): một scalar phân phối xác suốt thực tế
* Q(x): một scalar phân phối xác suốt theo mô hình dự đoán
* X: là các giá trị có thể xảy ra

Output:

Một giá trị đại diện cho sự chính xác của mô hình