HỌC VIỆN KỸ THUẬT QUÂN SỰ

**KHOA CÔNG NGHỆ THÔNG TIN**

****

# **BÁO CÁO HỌC PHẦN**

**TRÍ TUỆ NHÂN TẠO**

Đề tài: Nhận dạng đối tượng trong hình ảnh sử dụng bộ dữ liệu CIFAR-10

Giáo viên hướng dẫn: Ngô Hữu Phúc

Sinh viên thực hiện: Nguyễn Tiến Dương

Lớp: KHMT17

Mã sinh viên: 18150010

# **Cấu trúc mạng CNN**

* CNN (Mạng nơ-ron tích chập): là một trong những mô hình deep learning tiên tiến, nó giúp chúng ta trong việc xây dựng những hệ thống thông minh với độ chính xác cao như hiện nay.
* Những lớp cơ bản của mạng CNN:
  + Convolutional layer
  + ReLU layer
  + Pooling layer
  + Fully connected layer
* Convolutional layer

Đây là lớp quan trọng nhất của CNN, lớp này có nhiệm vụ thực hiện mọi tính toán. Lớp bao gồm các yếu tố: filter, stride, padding.

* + **filter**: CNN sử dụng các filter hay kernel để áp dụng trượt trong vùng của hình ảnh, thường có size là 3x3.
  + **stride**: có nghĩa là khi bạn dịch chuyển kernel trừ trái sang phải. Và sự chuyển dịch này chính là Stride.
  + **padding**: Là các giá trị 0 được thêm vào input image.
* ReLU layer
  + Relu layer là hàm kích hoạt trong neural network và hàm này còn được gọi là activation function.
  + Hiện nay, hàm relu được dùng phổ biến, nó được sử dụng nhiều cho việc huấn luyện mạng nơ-ron nó nhiều ưu điểm như tính toán sẽ trở nên nhanh hơn.
* Pooling layer
  + Lớp pooling thường được sử dụng ngay sau lớp convolutional để đơn giản hóa thông tin đầu ra, để làm giảm chất lượng của hình ảnh mà không làm mất đặc trưng của nó.
  + Pooling layer có 2 loại chủ yếu là: max pooling và average pooling.
* Fully connected
  + Lớp này có nhiệm vụ đưa ra kết quả sau khi lớp convolutional layer và pooling layer đã nhận được ảnh truyền. Lúc này, ta thu được kết quả là model đã đọc được thông tin của ảnh và để liên kết chúng cũng như cho ra nhiều output hơn thì ta sử dụng fully connected layer.

# **Giới thiệu về CIFAR-10**

* Bộ dữ liệu CIFAR-10 là một tập hợp các hình ảnh thường được sử dụng đào tạo các thuật toán Machine Learning và Computer Vision để nhận dạng các đối tượng. Vì hình ảnh trong CIFAR-10 có độ phân giải thấp cho phép thử nhanh các thuật toán để xem cái nào hoạt động.
* Bộ dữ liệu chứa 60.000 hình ảnh màu 32x32x3 ở 10 lớp khác nhau, đại diện cho: “máy bay, ô tô, chim, mèo, hươu, chó, ếch, ngựa, tàu và xe tải”.
* Bộ dữ liệu sẽ được chia ra 50.000 hình ảnh để train còn 10.000 hình ảnh để test.

# **Các bước xây dựng model**

## **source code**:

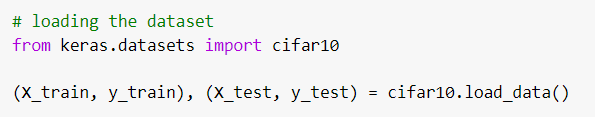
## <https://colab.research.google.com/drive/1et8M1bjhW0_ZQOD8fe-aYMp6ijoVUUam?usp=sharing>

## Pixels ảnh:

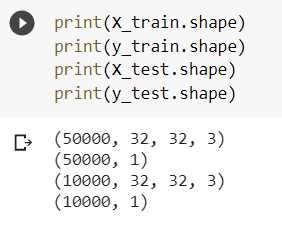
Được biểu diễn bởi 3 giá trị tương ứng với màu Red, Green, Blue và mỗi giá trị này sẽ nằm trong khoảng từ 0-255. Vì vậy trong bài toán này chúng ta coi mỗi pixel là một vector 3 chiều.

## **Bước 1: Tải lên bộ dữ liệu CIFAR-10**

Import bộ dữ liệu cifar-10 trong thư viện keras, sau đó chia bộ dữ liệu làm 2 bộ train và test:

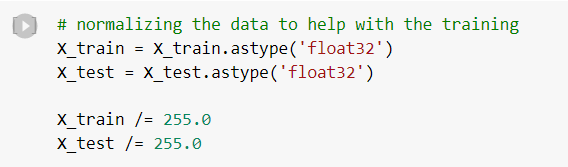


Bộ dữ liệu gồm 60.000 ảnh có size là 32x32x3, mặc định chia 50.000 ảnh để train và 10.000 ảnh còn lại để test:



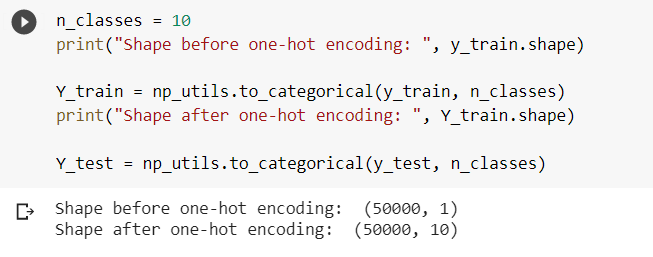
## Bước 2: Chuẩn hóa dữ liệu

Chuyển dữ liệu đầu vào về dạng float, sau đó chia 2 bộ dữ liệu cho 255 để việc tính toán trở nên nhanh hơn

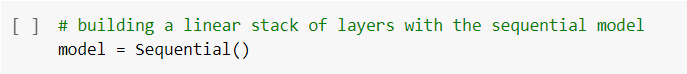


## **Bước 3: Mã** **hoá bộ dữ liệu train**

Chia dữ liệu trong bộ dữ liệu train ra làm 10 lớp đã được gán nhãn để phục vụ cho việc train dữ liệu:



## Bước 4: Khởi tạo model Sequential

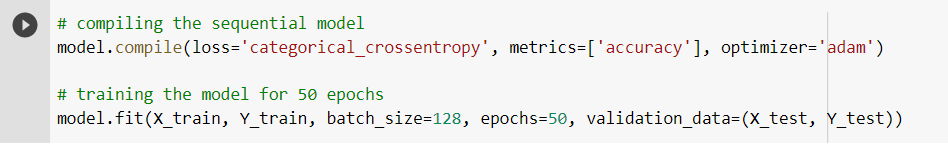
****

## Bước 5: Xây dựng các lớp của mạng

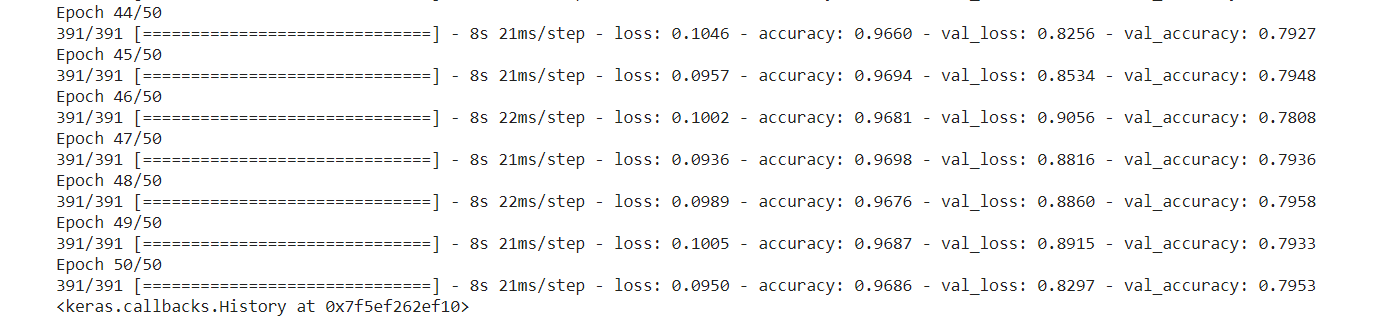
Sử dụng một mô hình mạng có sẵn, được cho là tối ưu nhất cho bài toán:

****

## Bước 6: Tiến hành train model

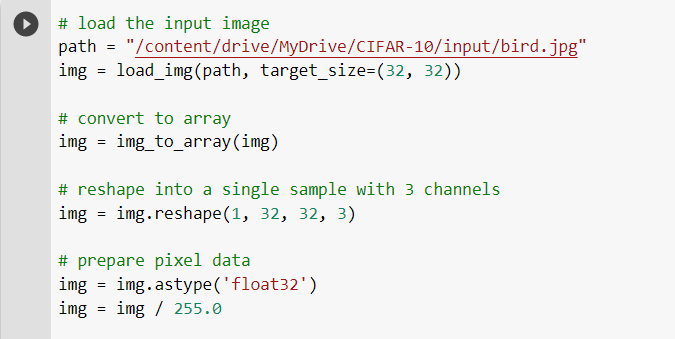
****

Kết quả:



# **Nhận diện đối tượng trong hình ảnh**

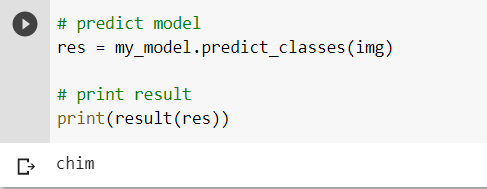
* Truyền vào một ảnh muốn nhận diện:



* Lấy hình ảnh chim làm ví dụ:



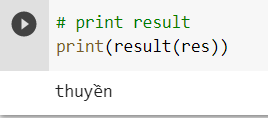
* Predict model, in kết quả ra màn hình:



* Lấy hình ảnh thuyền làm ví dụ:



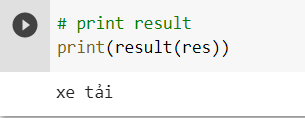
* Kết quả:



* Lấy hình ảnh xe tải làm ví dụ:



* Kết quả:



# **Nhận xét**

* Vẫn còn một số hạn chế ngoài việc độ chính xác chỉ đạt được gần 80%, đó là bộ dữ liệu dùng để train chỉ giới hạn ở 10 lớp chưa được mở rộng để có thể nhận diện được nhiều đối tượng hơn.
* Nhưng với một mô hình mạng CNN đơn giản, ta có thể áp dụng vào các bài toán thực tế. Mặc dù có thể kết quả chưa đạt cao, nhưng tiềm năng của nó đem lại vẫn là rất lớn nhờ sự đơn giản và nhanh chóng.