**HƯỚNG DẪN SỬ DỤNG**

***Dự án: Ứng dụng CNN để xây dựng hệ thống nhận diện chó mèo***

**Yêu cầu:**

* *Chạy chương trình trên PyCharm bằng máy tính*
* *Cấu hình máy tính:*

*+ Hệ điều hành: Windows 11, Windows 10, 8/8.1, Linux, Mac,…*

*+ CPU : Intel, Core i3 trở lên*

*+ RAM : 4 GB trở lên*

*+ Ổ cứng : HDD, SSD*

*+ System type: 64 bit, 32 bit*

**Bước 1**: Lên trang <https://www.python.org/downloads/> để cài đặt môi trường Python lên máy tính (Bản 3.13.\*)

A screenshot of a computer

Description automatically generated

Hình 1: Cài đặt môi trường chạy python.

**Bước 2**: Bật cmd và chạy lệnh dưới để cài Conda 

* **Lưu ý nếu bạn muốn giảm tải cho CPU thì cài CUDA, cuDNN** Mở Anaconda Prompt chạy



A black screen with a black border

Description automatically generated

Hình 2: Hình ảnh Anaconda Prompt.

**Bước 3**: Download source từ link và mở file Readme.md để xem các thư viện cần cài đặt cho chương trình.

A screenshot of a computer

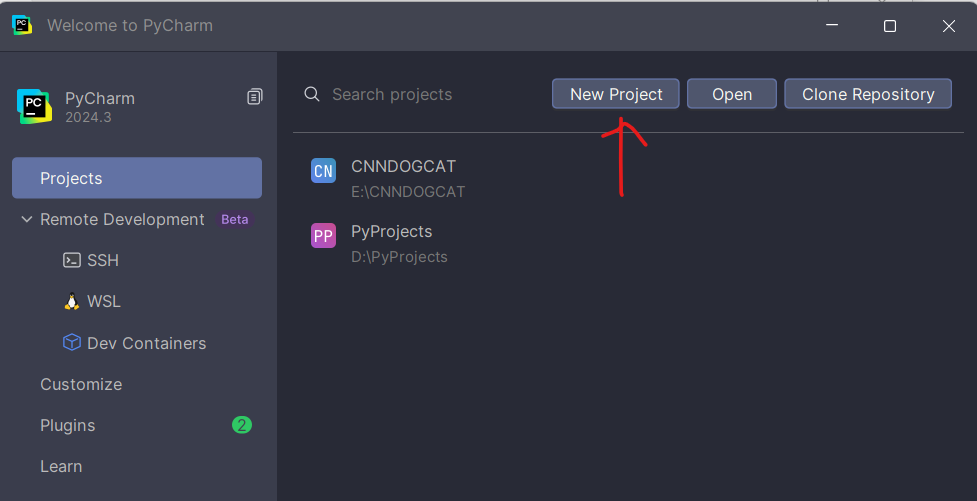
Description automatically generated

Hình 3: Danh sách các thư viện cần cài đặt.

* Có tổng cộng 6 thư viện cần cài đặt
* jax (bản từ 0.4.36 trở lên)
* keras (bản từ 3.7.0 trở lên)
* tensorflow (bản từ 2.18.0 trở lên)
* opencv-python (bản từ 4.10.0 trở lên)
* scikit-learn (bản từ 1.5.2 trở lên)
* matplotlib (bản từ 3.9.3 trở lên)

**Bước 4:** Cài PyCharm nếu chưa có và bật PyCharm:

* Chọn New Project (File->NewProject)



Hình 4: Lần đầu mở PyCharm.

* Cài đặt 2,3,4 như hình (1 là Vị trí đặt project nên tạo folder mới và đặt tên folder đó -> là tên project)

A screenshot of a computer program

Description automatically generated

Hình 5: Cài đặt môi trường Project.

**Bước 5:** Thao tác mở cmd ảo của project, chọn Command Prompt (Anaconda Prompt)

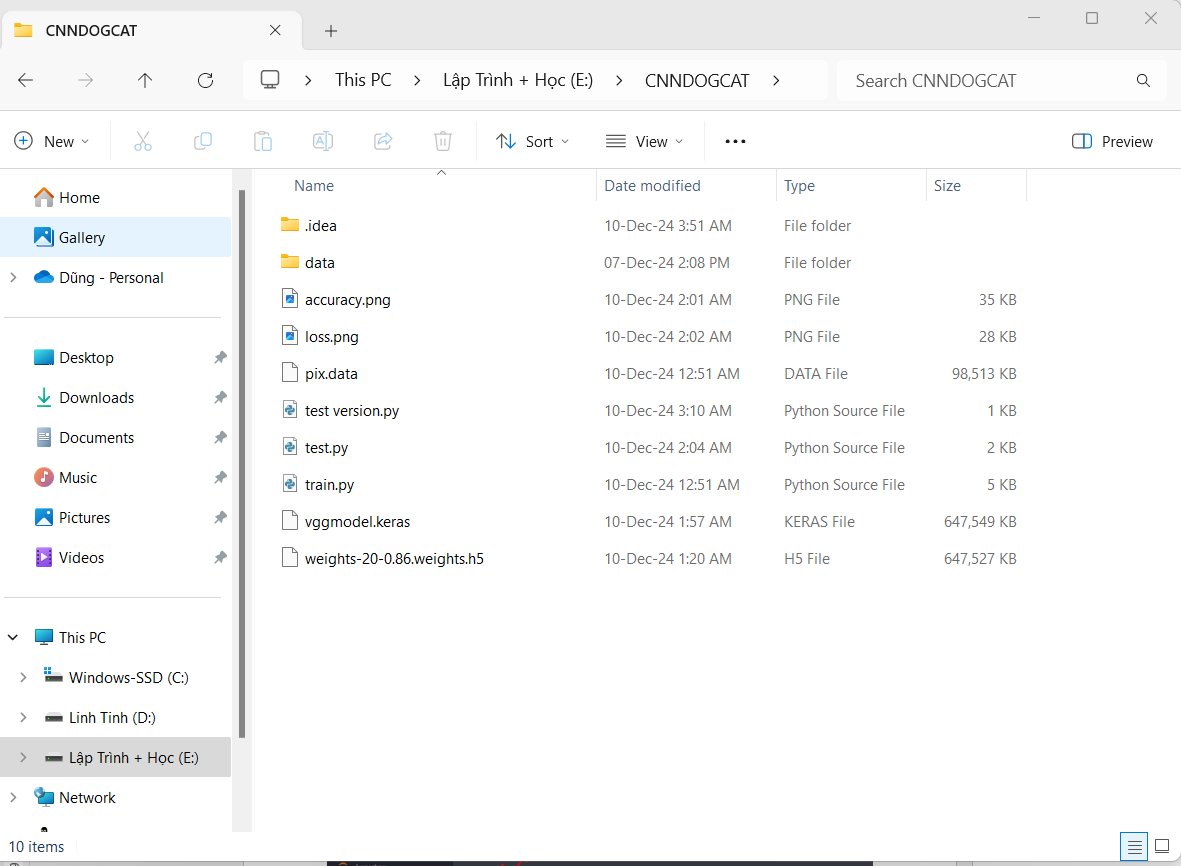
A screenshot of a computer

Description automatically generated

Hình 6: Cách cài đặt thư viện bằng pip trong cmd project

**Bước 6**: gõ cú pháp: pip install <tên\_thư\_viện> để cài các thư viện cần thiết trên terminal trên

**Bước 7:** Mở File Explorer và mở đến Project vừa tạo và copy tất cả source vào project bao gồm data và file .py



Hình 7: Thêm folder vào project.

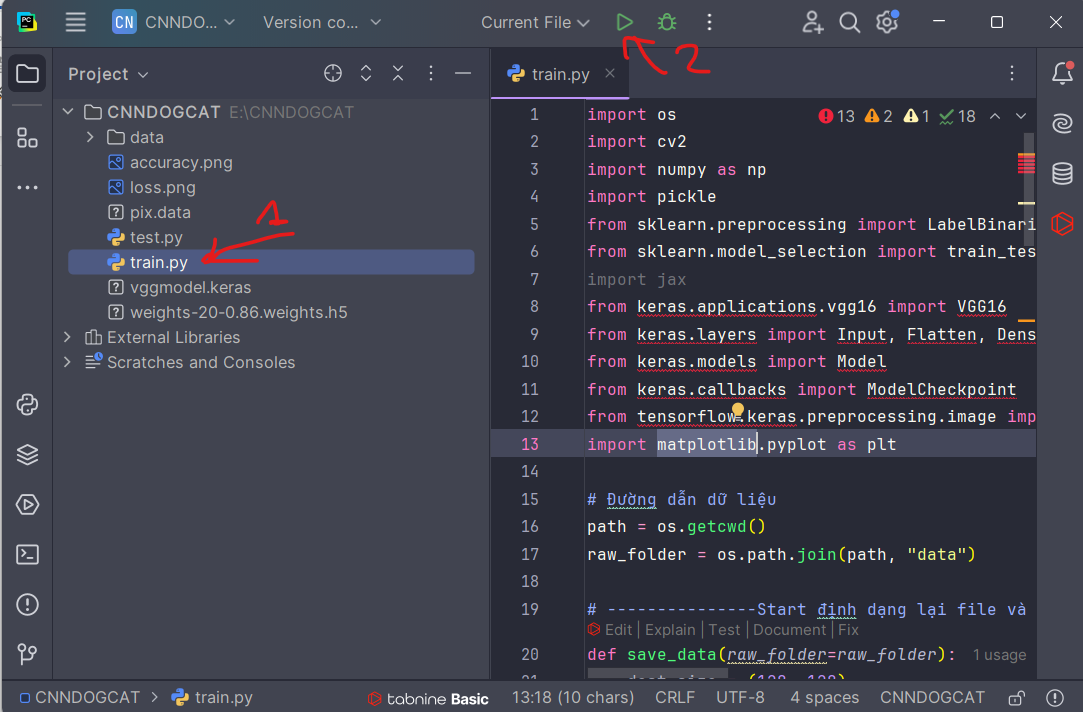
**Bước 8**: Quay lại PyCharm kiểm tra các folder đã được thêm vào chưa. Như dưới là OK.

A screenshot of a computer

Description automatically generated

Hình 8: Kiểm tra file trong PyCharm.

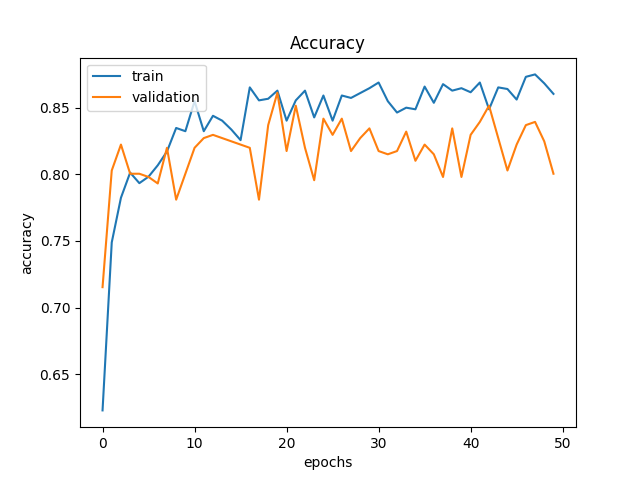
**Bước 9**: Mở file train.py và ấn vào hình tam giác để huấn luyện với bộ dữ liệu trong data.



Hình 9: Chạy ***train.py***

Sau khi chạy file ***train.py*** sẽ tạo ra 2 biểu đồ training loss và accuracy sau khi thành công.

\*Chú ý: Khi khởi chạy sẽ có 1 số thông báo sẽ không ảnh hưởng đến quá trình train.



Hình 10: Biểu đồ training và accuracy

A graph of a graph

Description automatically generated

Hình 11: Biểu đồ training loss

Từ biểu đồ training loss và accuracy, ta thấy ở những epoch đầu, tỷ lệ loss khá cao và sự chính xác vẫn chưa thực sự cao. Tuy nhiên ở epoch 1.0 trở đi, điều này đã được cải thiện rõ rệt mặc dù val\_loss vẫn chưa ổn định.

**Bước 10**: Sau khi chạy sẽ có nhiều checkpoint được ghi lại. Mở file ***test.py*** sau đó thay đổi trong nháy kép (3) bằng tên file .h5 có tỷ lệ lớn nhất, tỷ lệ của file ở vị trí (2) của tên. Sau đó thực hiện chạy chương trình bằng ấn 4.

A screenshot of a computer program

Description automatically generated

Hình 12: Chạy file ***test.py***

\*Lưu ý trước khi nhấn chạy hãy bật camera được kết nối với máy tính, và hãy đảm bảo chỉ duy nhất 1 cái hoạt động chính xác.

\*Sau khi chạy sẽ hiện thị lên cửa sổ hình ảnh hay đưa chó hoặc mèo để test

A screenshot of a computer

Description automatically generated

Hình 13: Chương trình trong khi chạy.

A screenshot of a computer

Description automatically generated

Hình 14: Chương trình trong khi chạy 2.