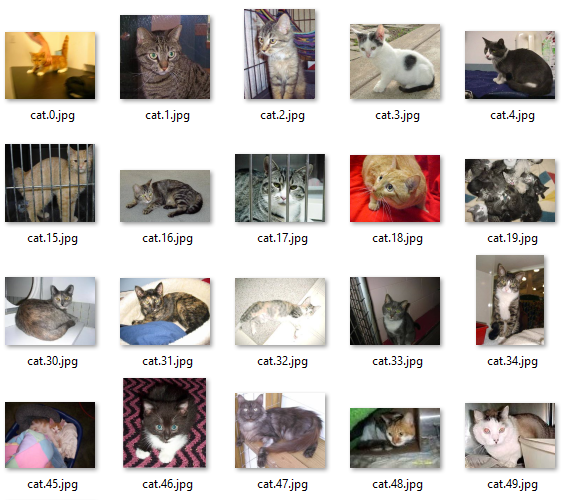
## **3. Bài toán phân loại ảnh chó mèo**

### ***3.1. Mô tả dữ liệu***

Bộ dữ liệu chó mèo tại nguồn: <https://storage.googleapis.com/mledu-datasets/cats_and_dogs_filtered.zip>

chứa hơn 5000 ảnh chó mèo màu khác nhau với hàng là 32, size ảnh là (160; 160)



|  |  |
| --- | --- |
| **label** | **class** |
| 0 | cat |
| 1 | dog |

### ***3.2. Mô tả bài toán học máy***

Bài toán sử dụng phân lớp nhị phân dựa vào đó phân loại ảnh chó mèo

Kỹ thuật sử dụng: Transfer learning và fine-tuning

### ***3.3. Mô tả bài toán học máy***

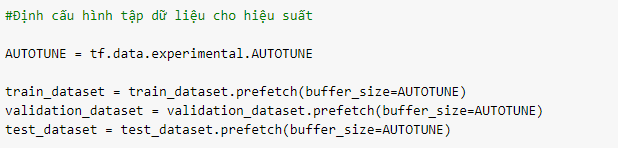


Mô hình được đào tạo trước là một mạng đã lưu đã được đào tạo trước đó trên một tập dữ liệu lớn, thường là trong một nhiệm vụ phân loại hình ảnh quy mô lớn. Bạn có thể sử dụng mô hình được đào tạo trước như hiện tại hoặc sử dụng học chuyển giao để tùy chỉnh mô hình này cho một nhiệm vụ nhất định.

Trực giác đằng sau việc học truyền để phân loại hình ảnh là nếu một mô hình được đào tạo trên một tập dữ liệu đủ lớn và tổng quát, thì mô hình này sẽ đóng vai trò là một mô hình chung của thế giới hình ảnh một cách hiệu quả. Sau đó, bạn có thể tận dụng các bản đồ đặc trưng đã học này mà không cần phải bắt đầu lại từ đầu bằng cách đào tạo một mô hình lớn trên một tập dữ liệu lớn.

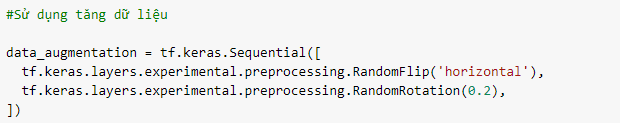
Định cấu hình tập dữ liệu cho hiệu suất

Sử dụng tìm nạp trước có bộ đệm để tải hình ảnh từ đĩa mà không bị chặn I / O. Để tìm hiểu thêm về phương pháp này, hãy xem hướng dẫn [hiệu suất dữ liệu](https://www.tensorflow.org/guide/data_performance) .



Sử dụng tăng dữ liệu

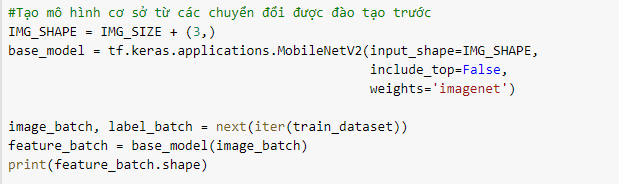
Khi bạn không có tập dữ liệu hình ảnh lớn, bạn nên đưa vào một cách giả tạo tính đa dạng mẫu bằng cách áp dụng các phép biến đổi ngẫu nhiên, nhưng thực tế cho hình ảnh huấn luyện, chẳng hạn như xoay và lật ngang. Điều này giúp mô hình tiếp xúc với các khía cạnh khác nhau của dữ liệu đào tạo và giảm việc trang bị [quá nhiều](https://www.tensorflow.org/tutorials/keras/overfit_and_underfit) . Bạn có thể tìm hiểu thêm về tăng dữ liệu trong [hướng dẫn](https://www.tensorflow.org/tutorials/images/data_augmentation) này .



Lưu ý: Nếu sử dụng hàm khác tf.keras.applications, hãy nhớ kiểm tra tài liệu API để xác định xem chúng có mong đợi pixel trong [-1,1]hoặc [0,1]hoặc sử dụng preprocess\_inputhàm được bao gồm hay không .

Tạo mô hình cơ sở từ các chuyển đổi được đào tạo trước

Bạn sẽ tạo mô hình cơ sở từ mô hình MobileNet V2 được phát triển tại Google. Điều này được đào tạo trước trên tập dữ liệu ImageNet, một tập dữ liệu lớn bao gồm 1,4 triệu hình ảnh và 1000 lớp. ImageNet là một tập dữ liệu đào tạo nghiên cứu với nhiều loại như jackfruitvà syringe. Cơ sở kiến ​​thức này sẽ giúp chúng tôi phân loại chó và mèo từ tập dữ liệu cụ thể của chúng tôi.

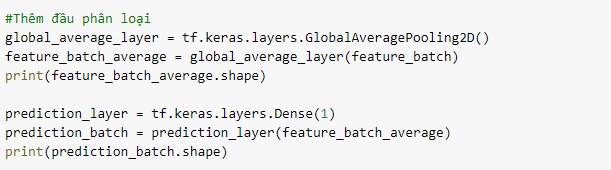


Khai thác tính năng

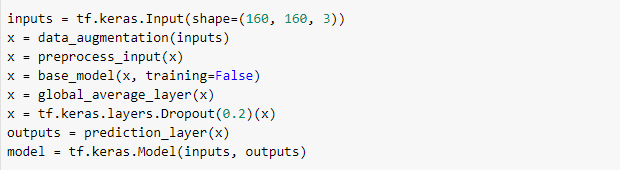
Trong bước này, bạn sẽ đóng băng cơ sở phức hợp được tạo từ bước trước và sử dụng như một trình trích xuất tính năng. Ngoài ra, bạn thêm một bộ phân loại lên trên nó và đào tạo bộ phân loại cấp cao nhất.

Thêm đầu phân loại

Để tạo dự đoán từ khối các đối tượng địa lý, hãy tính trung bình trên các 5x5vị trí không gian vùng, sử dụng một tf.keras.layers.GlobalAveragePooling2Dlớp để chuyển đổi các đối tượng địa lý thành một vectơ 1280 phần tử duy nhất trên mỗi hình ảnh.

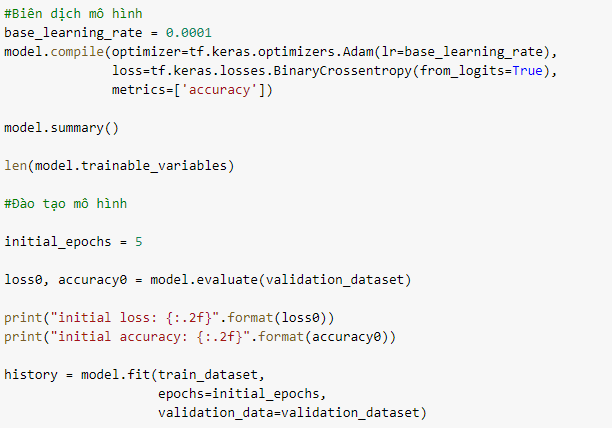


Xây dựng mô hình bằng cách xâu chuỗi các lớp tăng dữ liệu, thay đổi tỷ lệ, base\_model và tính năng với nhau bằng cách sử dụng [API chức năng Keras](https://www.tensorflow.org/guide/keras/functional) . Như đã đề cập trước đây, hãy sử dụng training = False vì mô hình của chúng tôi chứa một lớp BatchNormalization.



Biên dịch mô hình

Biên dịch mô hình trước khi đào tạo nó. Vì có hai lớp, hãy sử dụng suy hao entropy chéo nhị phân from\_logits=True Vì mô hình cung cấp đầu ra tuyến tính.



Đường cong học tập





Đánh giá và dự đoán

Cuối cùng, bạn có thể xác minh hiệu suất của mô hình trên dữ liệu mới bằng cách sử dụng bộ thử nghiệm.

# 