



TRƯỜNG ĐẠI HỌC
SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
HCMC University of Technology and Education

Phân loại rắn độc hay không độc và phân loại 8 loài rắn theo thời gian thực trên Mobile sử dụng thuật toán CNN

Sinh viên thực hiện: Lê Phan Văn Việt¹ – 19146302

¹ ĐH Sư Phạm Kỹ Thuật thành phố Hồ Chí Minh, 19146302@student.hcmute.edu.vn

Mô tả: Hiện nay có một số loài rắn sống gần môi trường sống của con người đặc biệt là vùng quê. Việc không biết rắn độc hay không dễ khiến con người gặp nguy hiểm đến tính mạng khi tiếp xúc với rắn độc. Trong đề tài nghiên cứu này, em đã phân loại rắn độc hay không độc bằng mô hình sử dụng thuật toán CNN (Mạng nơ-ron tích chập). Và cũng trong đề tài này em cũng phân loại 8 loài rắn. Cung cấp thông tin về loài rắn đó mức độ độc tính và giúp bác sĩ có thể kịp thời tìm ra huyết thanh cứu người. Để triển khai mô hình này, em đã sử dụng cơ sở dữ liệu được xây dựng bởi trình thu thập ảnh từ tổ hợp một số nguồn khác nhau với 9200 và 1800 hình ảnh tương ứng cho mỗi mô hình làm dữ liệu huấn luyện. Công việc bao gồm làm giàu dữ liệu, tiêu chuẩn hoá kích thước hình ảnh. Sau đó, mô hình được đào tạo và sau đó được kiểm tra để xác minh độ tin cậy của mô hình. Độ chính xác của mô hình rắn độc hay không độc là 64% và độ chính xác của mô hình còn lại là 60%. Để mô hình có tính linh động cao và thuận lợi thì mô hình được triển khai trên thiết bị di động có thể cho kết quả từ hình chụp hoặc ở chế độ realtime trên camera.

Từ khóa: trí tuệ nhân tạo; học máy; thuật toán Convolutional neural network; phân loại rắn độc hay không độc; phân loại một số loài rắn

1 Giới thiệu đề tài

Theo WHO trên thế giới có khoảng 5,4 triệu ca rắn cắn xảy ra mỗi năm, dẫn đến 1,8 đến 2,7 triệu ca nhiễm độc (ngộ độc do rắn cắn). Có từ 81 410 đến 137 880 trường hợp tử vong và số ca cắt cụt chân và các thương tật vĩnh viễn khác cao gấp ba lần mỗi năm^[1]. Một số nguyên nhân dẫn đến tỷ lệ tử vong và thương tật vĩnh viễn cao như là thiếu kiến thức không biết đó là rắn độc, đưa đến bệnh viện trễ, độc tính của rắn quá mạnh,... Trong đó việc thiếu kiến thức xác định rắn độc hay không để có phương án cứu chữa kịp thời là một nguyên nhân chính. Việc biết được đó có phải rắn độc hay không giúp mọi người có thể né tránh nguy hiểm và cũng như cứu chữa tránh những chủ quan đáng tiếc. Bên cạnh đó biết được thêm thông tin về loại rắn đó còn giúp mọi người xác định được mức độ nguy hiểm của loài và bác sĩ cũng nhanh chóng tìm ra huyết thanh phù hợp để cứu người. Vì vậy việc xây dựng lên mô hình có thể nhận diện phân loại rắn độc hay không độc cũng như nhận diện các loài rắn là cần thiết.

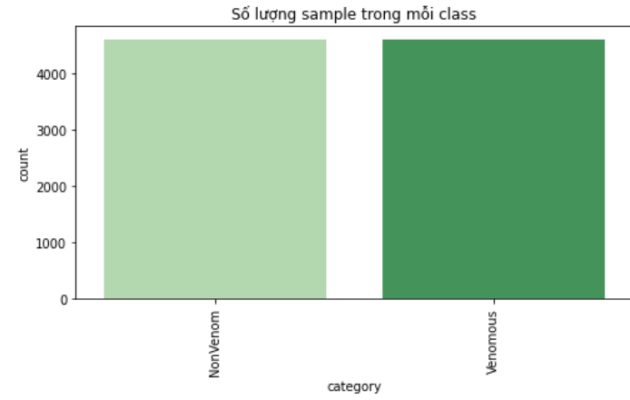
2 Dữ liệu và phương pháp nghiên cứu

Phần này trình bày phương pháp trong được thực hiện trong đề tài, bao gồm các chi tiết của tập dữ liệu thử nghiệm, các bước xử lý trước dữ liệu được thực hiện để thiết lập các thử nghiệm và phương pháp được áp dụng để giải quyết các mục đích nghiên cứu.

2.1 Tập dữ liệu

2.1.1 Mô hình phân loại rắn độc hay không độc

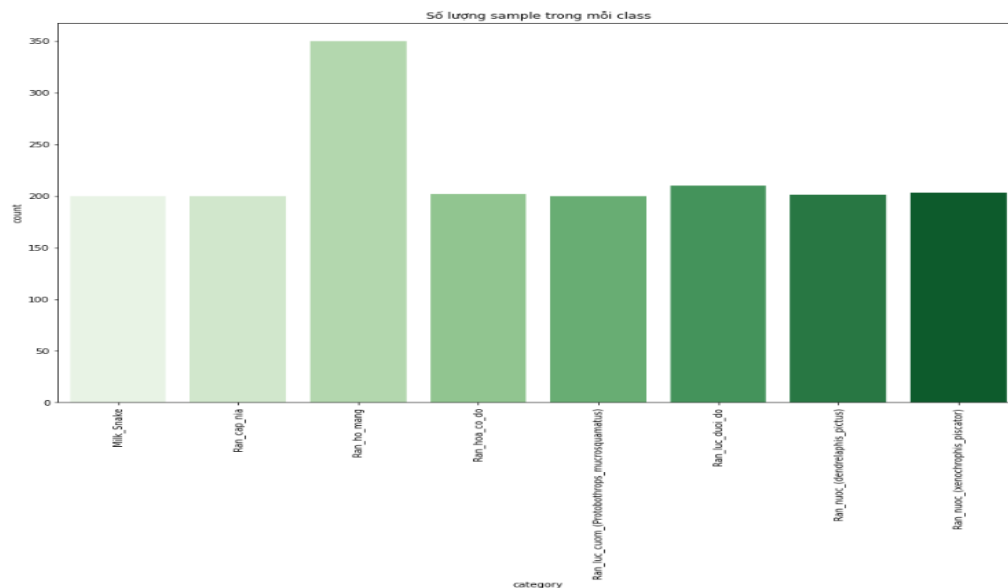
Tập dữ liệu dùng để huấn luyện mô hình được em tự thu thập chủ yếu từ google image và một số trang web khác. Tập dữ liệu bao gồm 9200 chia đều cho 2 lớp: rắn độc và rắn không độc. Tất cả dữ liệu hình ảnh tỷ lệ 1:1 và độ phân giải lớn nhất là 1417 pixel, thấp nhất là 130 pixel.



Hình 1 Số lượng sample mỗi class của mô hình phân loại rắn độc hay không độc

2.1.2 Mô hình phân loại 8 loại rắn

Tập dữ liệu bao gồm 8 loại rắn: Milk snake, rắn cạp nia, rắn hổ mang, rắn lục đuôi đỏ, rắn cỏ hoa cổ đỏ, rắn lục cườm, rắn nước (Dendrelaphis pictus) và rắn nước (xenochrophis piscator). Tập dữ liệu dùng để huấn luyện mô hình được em tự thu thập chủ yếu từ google image và một số trang web khác. Tập dữ liệu bao gồm gần 1800 Các lớp có số lượng dữ liệu nằm ở khoảng 200 tấm. Số lượng dữ liệu nhiều nhất là Rắn hổ mang với 350 tấm. Tất cả dữ liệu hình ảnh tỷ lệ 1:1 và độ phân giải lớn nhất là 1417 pixel, thấp nhất là 130 pixel.



Hình 2 Số lượng sample mỗi class của mô hình phân loại 8 loài rắn

2.1.3 Xử lý dữ liệu

Tập dữ liệu có kích thước nhỏ sẽ ảnh hưởng đến hiệu suất của mô hình. Vậy nên cần tăng kích thước lên bằng kỹ thuật tăng cường hình ảnh. Tăng cường hình ảnh là một kỹ thuật hữu ích được sử dụng để tăng tính đa dạng của tập dữ liệu huấn luyện sao cho các bản sao thực tế nhưng ngẫu nhiên của hình ảnh gốc có thể được tạo ra thông qua các phép biến đổi đơn giản như thay đổi hình học và không gian màu, cắt hình ảnh, chèn nhiễu và xóa ngẫu nhiên. Trong đề tài này em sử dụng ImageDataGenerator được cung cấp bởi thư viện Keras để tăng cường dữ liệu hình ảnh.

Hàm	Thông số
horizontal_flip	True
vertical_flip	True
Rescale	1. / 255
rotation_range	45
brightness_range	0.5 - 1.0
zoom_range	0.5 - 1.0
width_shift_range	0.2
height_shift_range	0.2

Ảnh được tải lên sẽ được định dạng ở dải màu RGB và kích thước 100x100 ở mô hình phân loại rắn độc hay không độc. Còn ở mô hình hình phân loại loài rắn Ảnh được tải lên sẽ được định dạng ở dải màu RGB và kích thước 100x100

2.2 CNN mạng nơ-ron tích chập

Convolutional Neural Networks (CNN) là một trong những mô hình deep learning phổ biến nhất và có ảnh hưởng nhiều nhất trong cộng đồng Computer Vision. CNN được dùng trong trong nhiều bài toán như nhận dạng ảnh, phân tích video, ảnh MRI, hoặc cho bài

các bài của lĩnh vực xử lý ngôn ngữ tự nhiên, và hầu hết đều giải quyết tốt các bài toán này^[2].

CNN bao gồm tập hợp các lớp cơ bản bao gồm: convolution layer + nonlinear layer, pooling layer, fully connected layer. Các lớp này liên kết với nhau theo một thứ tự nhất định. Thông thường, một ảnh sẽ được lan truyền qua tầng convolution layer + nonlinear layer đầu tiên, sau đó các giá trị tính toán được sẽ lan truyền qua pooling layer, bộ ba convolution layer, nonlinear layer và pooling layer có thể được lặp lại nhiều lần trong network. Và sau đó được lan truyền qua tầng fully connected layer và softmax để tính xác suất ảnh đó chứa vật thể gì.

Mô hình Phân loại rắn độc hay không độc sử dụng thuật toán CNN có số lớp tích chập là 4 và số lớp đầy đủ là 3. Mỗi lớp tích chập có lớp tổng hợp riêng và lớp chuẩn hóa riêng có kích thước đồng nhất là 2x2 và bước là 2. Ngoài ra, tất cả lớp tích chập đều áp dụng hàm "Relu" để khử tuyến tính và có đều có kernel = 3x3. Khả năng mỗi hình ảnh nhận dạng chính xác cho mỗi mẫu được tính toán dựa trên bộ phân loại "sigmoid"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 100, 100, 32)	896
max_pooling2d (MaxPooling2D)	(None, 50, 50, 32)	0
conv2d_1 (Conv2D)	(None, 50, 50, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 25, 25, 64)	0
flatten (Flatten)	(None, 40000)	0
dense (Dense)	(None, 128)	5120128
dense_1 (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 2)	258

Mô hình Phân loại rắn độc hay không độc sử dụng thuật toán CNN có số lớp tích chập là 4 và số lớp đầy đủ là 3. Mỗi lớp tích chập có lớp tổng hợp riêng và lớp chuẩn hóa riêng có

kích thước đồng nhất là 2x2 và bước là 2. Ngoài ra, tất cả lớp tích chập đều áp dụng hàm "Relu" để khử tuyến tính và có đều có kernel = 3x3. Khả năng mỗi hình ảnh nhận dạng chính xác cho mỗi mẫu được tính toán dựa trên bộ phân loại "softmax"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 50, 50, 32)	896
max_pooling2d (MaxPooling2D)	(None, 25, 25, 32)	0
conv2d_1 (Conv2D)	(None, 25, 25, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 12, 12, 64)	0
flatten (Flatten)	(None, 9216)	0
dense (Dense)	(None, 128)	1179776
dense_1 (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 8)	1032

Cả hai mô hình đều dùng chọn thuật toán huấn luyện là adam, và sử dụng thuật toán earlyStopping để tìm ra điểm hội tụ tránh overfitting.

2.3 Đánh giá mô hình

Sử dụng đồ thì training curve, Ma trận nhầm lẫn (confusion matrix), learning curve, precision, recall, f1 score để đánh giá mô hình.

2.4 RealTime

Để mô hình có thể sử dụng một cách thuận lợi đối với mọi người thì nó được triển khai trên ứng dụng điện thoại. Ứng dụng điện thoại được hướng đến ở đây là ứng dụng điện thoại Android được viết trên ngôn ngữ JAVA. Hiện tại [TensorFlow](#) (thư viện mã nguồn mở cho machine learning) có phiên bản hỗ trợ triển khai mô hình học máy lên thiết bị Android.

[TensorFlow](#) là một nền tảng mã nguồn mở “end-to-end” (nghĩa là tất cả trong một) của Nhóm Google Brain. TensorFlow là một thư viện phần mềm mã nguồn mở cho phép thực hiện các tác vụ máy học^[3].

TensorFlow Lite là phiên bản rút gọn của TensorFlow. Đây chính là phiên bản dùng để hỗ trợ triển khai mô hình học máy trên ứng dụng di động. Đặc điểm của TensorFlow Lite

- Nhanh hơn, do nó cho phép thực hiện machine learning ngay trên mobile với độ trễ thấp.
- Tốn ít dung lượng nên khá tốt cho mobile.

Để triển khai mô hình sử dụng thuật toán CNN cần chuyển đổi file h5 sang file tflite để có thể sử dụng kết quả huấn luyện mô hình trên TensorFlow Lite. Sau khi chuyển đổi file xong thì nhúng file đó vào trong Android Studio. Ở đây vì cả hai đều thuộc sở hữu của Google nên trên Android Studio có hỗ trợ trực tiếp việc nhúng file tflite vào ứng dụng.

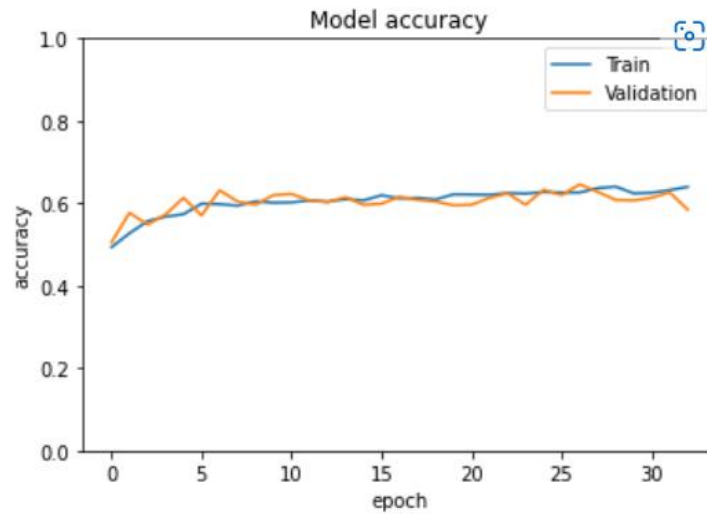
Ứng dụng điện thoại sau khi được nhúng file tflite có thể phân loại rắn độc hay không độc và phân loại 8 loại rắn trực tiếp thông qua camera hoặc có thể chụp hình lấy ảnh phân loại. Trước khi tiến hành phân loại thì ảnh cần được xử lý chuyển về dạng Bitmap. Ở dạng Bitmap cho phép chuẩn hoá hình ảnh để phù hợp với mô hình (Chuyển đổi kích thước ảnh, chuyển đổi dải màu RGB)



Hình 3 Triển khai mô hình lên ứng dụng di động

3 Kết quả

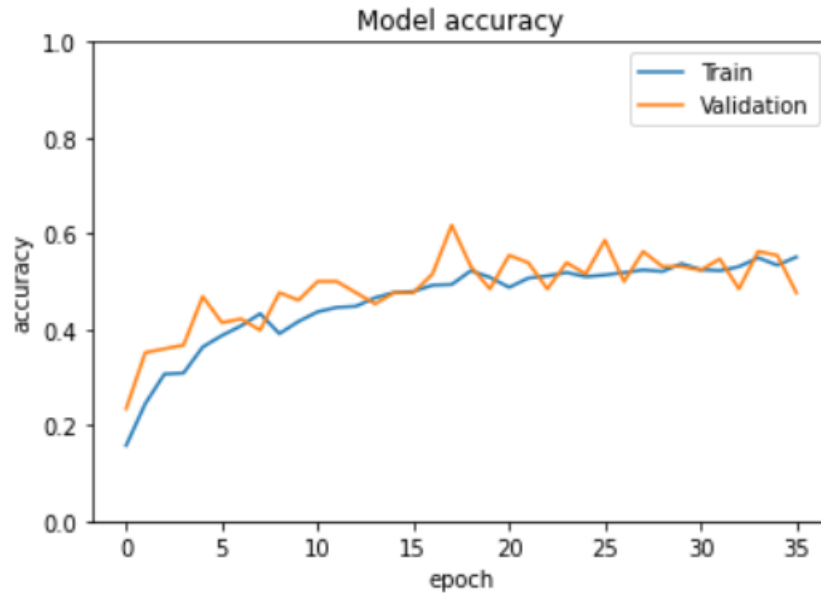
Sau khi huấn luyện mô hình độ chính xác thu được ở mô hình phân loại rắn độc hay không độc là 64% và mô hình phân loại 8 loại rắn là 60%. Cả 2 mô hình đều không bị overfitting



Hình 4 Đồ thị training curve của mô hình phân loại rắn độc hay không độc

Classification Report				
	precision	recall	f1-score	support
NonVenom	0.64	0.65	0.64	457
Venomous	0.65	0.63	0.64	463
accuracy			0.64	920
macro avg	0.64	0.64	0.64	920
weighted avg	0.64	0.64	0.64	920

Hình 5 Kết quả F1 score, pericision và recall của mô hình phân loại rắn độc hay không độc



Hình 6 Đồ thị training curve của mô hình phân loại 8 loài rắn

Classification Report

	precision	recall	f1-score	support
Milk_Snake	0.91	0.91	0.91	23
Ran_cap_nia	0.50	0.79	0.61	19
Ran_ho_mang	0.55	0.49	0.52	35
Ran_hoa_co_do	0.60	0.46	0.52	13
Ran_luc_cuom_(Protobothrops_mucrosquamatus)	0.18	0.36	0.24	14
Ran_luc_duoi_do	0.83	0.91	0.87	22
Ran_nuoc_(dendrelaphis_pictus)	0.87	0.45	0.59	29
Ran_nuoc_(xenochrophis_piscator)	0.33	0.24	0.28	21
accuracy			0.58	176
macro avg	0.60	0.58	0.57	176
weighted avg	0.63	0.58	0.58	176

Hình 7 Kết quả F1 score, pericision và recall của mô hình phân loại 8 loài rắn

Mô hình được triển khai trên ứng dụng điện thoại có thể phân loại dựa vào ảnh trực tiếp hoặc có thể phân loại trực tiếp trên camera.

4 Kết luận

Mô hình hoạt động tốt trên ứng dụng. Nhưng về độ chính xác cần cải thiện để có thể phân loại tốt hơn. Thông qua bảng F1 score để bổ sung thêm dữ liệu cho các lớp có phần trăm thấp và tập dữ liệu cần được xử lý mất cân đối. Thay đổi một vài thông số trong CNN để cải thiện.

Trong tương lai khi độ chính xác được cải thiện thì mô hình cần được huấn luyện để có thể phân loại ở khoảng cách xa hơn tránh nguy hiểm cho người sử dụng, và thêm mô hình phân loại mức độ độc tính.

Tài liệu tham khảo

1. “Snakebite envenoming”, WHO, <https://www.who.int>
2. “Tìm Hiểu Convolutional Neural Networks Cho Phân Loại Ảnh”, Quốc Thịnh, <https://pbcquoc.github.io/cnn/?fbclid=IwAR1zP7AnsFEYCYUfPFiirvV6lcQF4FpLEvmLMHPQzX5pl6X RD5ICMf78I>
3. “Sử dụng TensorFlow Lite Library để nhận diện Object”, Lê Minh Quang <https://viblo.asia/p/android-tensorflow-lite-machine-learning-example-924IjXmKPM>