

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
KHOA ĐIỆN ĐIỆN TỬ  
KHOA CÔNG NGHỆ THÔNG TIN



**HCMUTE**

**MÔN HỌC: ĐIỆN TOÁN ĐÁM MÂY**  
**ĐỀ TÀI: MACHINE LEARNING MODEL TRAINING AND  
DEPLOYMENT USING AN AWS EC2 INSTANCE**

**GVHD : TS. HUỖNH XUÂN PHỤNG**

**SINH VIÊN THỰC HIỆN:**

NGUYỄN ĐẶNG MAI THY      20139039

Thành phố Hồ Chí Minh, Tháng 12 năm 2023

## **LỜI CAM ĐOAN**

Em xin cam kết đồ án cuối kì này là đề tài do chính em nghiên cứu và thực hiện. Em chỉ dựa trên mà không sao chép từ bất kì tài liệu hay công trình nào đã được thực hiện trước đó. Nếu có bất kỳ vi phạm nào, em xin hoàn toàn chịu trách nhiệm.

Người thực hiện

**Nguyễn Đặng Mai Thy**

## LỜI CẢM ƠN

Trước khi đi vào nội dung của báo cáo, em xin chân thành cảm ơn Trường Đại học Sư Phạm Kỹ Thuật Thành phố Hồ Chí Minh, cảm ơn khoa Công Nghệ Thông Tin đã tạo ra môn học Điện toán đám mây để em cũng như toàn thể sinh viên nói chung có thể tiếp cận được những kiến thức hữu ích.

Tiếp theo, không thể quên gửi lời cảm ơn chân thành đến người đã giúp em có được sự tự tin về kiến thức để thực hiện được đề tài lần này, thầy Huỳnh Xuân Phụng. Trong suốt 15 tuần học, thầy là người trực tiếp chỉ dẫn để em có thể gạt hái được kiến thức về điện toán đám mây nhanh chóng và dễ dàng hơn.

Mặc dù em chỉ mới gặp thầy trong một vài tháng nhưng em đã cảm nhận được sự tận tình của thầy trong việc chỉ bảo, định hướng và hỗ trợ cho em cả về vật chất và tinh thần.

Tuy em đã nỗ lực trong suốt quá trình học tập nhưng việc còn tồn tại những thiếu sót về mặt kiến thức là không thể tránh khỏi. Những thiếu sót ấy có thể được hiện hữu trong đồ án của em. Nếu bắt gặp bất cứ sai sót nào, mong quý thầy cô cũng như các bạn có thể phản hồi cho em, em luôn vui lòng ghi nhận lấy những thiếu sót của mình và khắc phục nó nhanh nhất có thể.

Em mong rằng sẽ nhận được sự góp ý, bổ sung để đề tài của em được hoàn thiện nhất có thể và rút ra những kinh nghiệm cho những môn học tiếp theo.

Lời cuối cùng, em kính chúc thầy luôn mạnh khỏe, hạnh phúc, thành công trong công cuộc giảng dạy của mình.

# Mục lục

LỜI CẢM ƠN.....	3
CHƯƠNG 1 .....	6
1.1    GIỚI THIỆU ĐỀ TÀI.....	6
1.2    MỤC TIÊU ĐỀ TÀI.....	7
1.3    PHẠM VI ĐỀ TÀI .....	7
CHƯƠNG 2 .....	9
2.1    KHÁI NIỆM .....	9
A.    EC2 Instance.....	9
B.    Key Pairs, Private Key.....	9
C.    Machine Learning Model .....	10
D.    Classification.....	11
2.2    CÁC NGHIÊN CỨU LIÊN QUAN .....	11
CHƯƠNG 3 .....	13
3.1    CÁC CÔNG CỤ, CÔNG NGHỆ SỬ DỤNG .....	13
A.    Amazon Web Services ( AWS ).....	13
B.    Postman.....	14
C.    WinSCP.....	15
D.    PuTTY.....	16
E.    Operating System.....	17
F.    Development Environment.....	18
G.    Phát triển ứng dụng Web.....	19
H.    Machine Learning Model.....	21
3.2    THIẾT KẾ KIẾN TRÚC ĐỀ TÀI .....	23
3.3    CÁC BƯỚC THỰC HIỆN ĐỀ TÀI.....	24
CHƯƠNG 4 .....	36
4.1    NHỮNG THÁCH THỨC VÀ GIẢI PHÁP .....	36
4.2    QUẢN LÝ PHIÊN BẢN VÀ HỢP TÁC.....	38
CHƯƠNG 5 .....	40
5.1    KIỂM TRA CHỨC NĂNG .....	40
5.2    SỐ LIỆU HIỆU SUẤT.....	41
CHƯƠNG 6 .....	44

<b>6.1</b>	<b>THÔNG TIN CHI TIẾT VỀ DỊCH VỤ .....</b>	<b>44</b>
<b>6.2</b>	<b>PHÂN TÍCH CHI PHÍ.....</b>	<b>45</b>
	<b>CHƯƠNG 7 .....</b>	<b>46</b>
<b>7.1</b>	<b>KẾT QUẢ ĐẠT ĐƯỢC.....</b>	<b>46</b>
<b>7.2</b>	<b>PHÁT TRIỂN TRONG TƯƠNG LAI .....</b>	<b>48</b>
	<b>TÀI LIỆU THAM KHẢO.....</b>	<b>49</b>
	<b>PHỤ LỤC.....</b>	<b>50</b>

# CHƯƠNG 1

## TỔNG QUAN

### 1.1 GIỚI THIỆU ĐỀ TÀI

Đề tài "Đào tạo và triển khai mô hình học máy bằng cách sử dụng một EC2 Instance trên AWS" dẫn đầu bởi sự phát triển nhanh chóng của học máy và tính toán đám mây, đã mở ra nhiều cơ hội để huấn luyện và triển khai mô hình học máy trên nền tảng điện toán đám mây. Trong đề tài này, chúng ta tập trung vào việc nghiên cứu và thực hiện quá trình huấn luyện và triển khai mô hình học máy bằng cách sử dụng một EC2 (Elastic Compute Cloud) Instance trên AWS (Amazon Web Services).

EC2 là một dịch vụ cung cấp máy ảo linh hoạt trên đám mây của AWS. Nó cung cấp khả năng tăng giảm tự động và linh hoạt về tài nguyên máy chủ, cho phép chúng ta dễ dàng thích ứng với yêu cầu của quá trình đào tạo và triển khai mô hình học máy. Chúng ta sẽ tận dụng khả năng mở rộng và khả năng tính toán mạnh mẽ của EC2 để tăng tốc độ đào tạo mô hình và triển khai nó vào một môi trường thực tế.

Trong quá trình nghiên cứu, chúng ta sẽ khám phá các phương pháp, kỹ thuật và công nghệ hiện đại trong việc huấn luyện mô hình học máy. Điều này bao gồm việc tiền xử lý dữ liệu, chọn mô hình phù hợp, điều chỉnh siêu tham số, và đánh giá hiệu suất mô hình. Chúng ta cũng sẽ tìm hiểu về các thư viện và công cụ hỗ trợ như TensorFlow để xây dựng và huấn luyện mô hình.

Sau khi hoàn thành quá trình huấn luyện, chúng ta sẽ tập trung vào việc triển khai mô hình đã huấn luyện vào một môi trường thực tế. Sử dụng EC2 Instance, chúng ta sẽ tạo và cấu hình một môi trường phù hợp để triển khai mô hình vào một ứng dụng hoặc hệ thống. Chúng ta sẽ khám phá các phương pháp triển khai, bao gồm triển khai mô hình trên máy chủ web, ứng dụng di động, hoặc tích hợp vào các hệ thống sẵn có.

Đề tài này không chỉ mang lại sự hiểu biết sâu sắc về quy trình huấn luyện và triển khai mô hình học máy, mà còn nắm vững việc sử dụng EC2 Instance trên AWS. Nó cung cấp cơ hội cho chúng ta để làm việc với các công nghệ hàng đầu trong lĩnh vực học máy và điện toán đám mây, và áp dụng chúng vào việc giải quyết các vấn đề thực tế.

## 1.2 MỤC TIÊU ĐỀ TÀI

Mục tiêu chính của đề tài "Đào tạo và triển khai mô hình học máy bằng cách sử dụng một EC2 Instance trên AWS" là:

- Nghiên cứu và hiểu rõ về quá trình huấn luyện mô hình học máy: Đề tài sẽ tập trung vào việc khám phá các phương pháp, kỹ thuật và công nghệ hiện đại trong việc huấn luyện mô hình học máy. Mục tiêu là hiểu rõ quy trình từ tiền xử lý dữ liệu, chọn mô hình, điều chỉnh siêu tham số và đánh giá hiệu suất mô hình.
- Tìm hiểu và sử dụng dịch vụ EC2 trên AWS: Mục tiêu là làm quen với EC2 (Elastic Compute Cloud) Instance trên AWS và tận dụng khả năng mở rộng và tính toán mạnh mẽ của nó để tăng tốc quá trình đào tạo mô hình học máy.
- Triển khai mô hình học máy vào một môi trường thực tế: Mục tiêu là tạo và cấu hình một môi trường phù hợp sử dụng EC2 Instance để triển khai mô hình đã huấn luyện vào một ứng dụng hoặc hệ thống thực tế. Chúng ta sẽ khám phá các phương pháp triển khai, bao gồm triển khai trên máy chủ web, ứng dụng di động hoặc tích hợp vào các hệ thống sẵn có.
- Áp dụng kiến thức và công nghệ vào giải quyết các vấn đề thực tế: Mục tiêu cuối cùng là áp dụng kiến thức và công nghệ thu được để giải quyết các vấn đề thực tế trong lĩnh vực học máy và triển khai mô hình. Điều này có thể bao gồm phân loại dữ liệu, dự đoán hoặc xử lý ảnh, hay bất kỳ ứng dụng nào khác có liên quan đến học máy.

Tóm lại, đề tài này nhằm mục đích cung cấp kiến thức về quá trình huấn luyện và triển khai mô hình học máy, đồng thời khám phá và sử dụng dịch vụ EC2 trên AWS để nâng cao khả năng và hiệu suất của quá trình này.

## 1.3 PHẠM VI ĐỀ TÀI

Phạm vi của đề tài "Đào tạo và triển khai mô hình học máy bằng cách sử dụng một EC2 Instance trên AWS" là tận dụng các dịch vụ trong môi trường AWS như EC2, cùng với các ứng dụng PuTTY, WinSCP và Streamlit, để xây dựng một ứng dụng web. Cụ thể, dự án bao gồm các hoạt động sau:

- Triển khai EC2: Tạo và cấu hình một EC2 Instance trên AWS để xây dựng cơ sở hạ tầng cho ứng dụng web.
- Sử dụng PuTTY và WinSCP: Sử dụng PuTTY để thiết lập kết nối SSH với

EC2 Instance và quản lý từ xa. Sử dụng WinSCP để truyền tải các tệp tin và thư mục giữa máy cục bộ và EC2 Instance.

- Xây dựng ứng dụng web: Sử dụng các framework và thư viện phù hợp trên EC2 Instance để phát triển ứng dụng web. Sử dụng Streamlit để xây dựng giao diện người dùng tương tác và hiển thị kết quả từ mô hình học máy.

Kết quả cuối cùng của dự án là một ứng dụng web có khả năng triển khai và chạy trên EC2 Instance, cho phép người dùng tương tác với mô hình học máy và xem kết quả một cách trực quan.



## CHƯƠNG 2

# CƠ SỞ LÝ LUẬN

### 2.1 KHÁI NIỆM

#### A. EC2 Instance

EC2 Instance là một dịch vụ của Amazon Web Services (AWS) cho phép người dùng thuê và quản lý các máy ảo trong đám mây. EC2 (Elastic Compute Cloud) cung cấp khả năng mở rộng linh hoạt, dễ dàng triển khai và quản lý các máy ảo trên nền tảng điện toán đám mây của AWS.

Mỗi EC2 Instance tương đương với một máy tính ảo độc lập, có tài nguyên phần cứng và hệ điều hành riêng. Người dùng có thể chọn loại EC2 Instance phù hợp với nhu cầu của họ, bao gồm loại CPU, bộ nhớ, lưu trữ và hiệu năng mạng.

EC2 Instance cho phép người dùng cài đặt và chạy các ứng dụng, hệ điều hành và dịch vụ trên máy ảo của mình. Nó cung cấp khả năng linh hoạt trong việc tăng giảm quy mô máy ảo dựa trên yêu cầu và tài nguyên thực tế. Người dùng có thể quản lý EC2 Instance thông qua giao diện điều khiển AWS hoặc API của AWS.

EC2 Instance là một công cụ quan trọng cho việc xây dựng và triển khai ứng dụng, xử lý phân tán, phân tích dữ liệu và nhiều tác vụ khác trên đám mây của AWS.

#### B. Key Pairs, Private Key

- Key Pairs (Cặp khóa): Trong EC2, Key Pairs được sử dụng để cung cấp quyền truy cập an toàn vào các instances EC2. Key Pairs trong EC2 bao gồm một cặp khóa gồm public key (khóa công khai) và private key (khóa riêng). Khi tạo một instance EC2, bạn có thể chọn một Key Pair để gắn kết với instance đó.
- Public key: Public key được sử dụng để mã hóa dữ liệu và cho phép truy cập vào instance EC2 thông qua giao thức SSH (Secure Shell). Public key có thể được chia sẻ với bất kỳ ai mà bạn muốn truy cập vào instance EC2 của bạn.
- Private key: Private key là một tập tin bảo mật và được giữ bí mật. Nó được sử dụng để giải mã dữ liệu được mã hóa bằng public key và xác thực người

dùng khi kết nối vào instance EC2 thông qua SSH. Private key không nên được tiết lộ cho bất kỳ ai ngoại trừ chính người quản lý instance EC2.

Khi kết hợp Key Pair với instance EC2, bạn có thể sử dụng private key tương ứng để truy cập vào instance bằng cách kết nối SSH. Điều này đảm bảo rằng chỉ những người có private key mới có thể truy cập vào instance và thực hiện các hoạt động quản lý và cấu hình.

### **C. Machine Learning Model**

Machine Learning Model (Mô hình học máy) là một thuật toán hoặc một tập hợp các thuật toán được sử dụng để xác định một quy luật hoặc mô hình dự đoán từ dữ liệu huấn luyện. Mô hình học máy học từ dữ liệu và tự điều chỉnh để đưa ra dự đoán hoặc phân loại cho dữ liệu mới.

Mô hình học máy có thể được áp dụng cho nhiều vấn đề và lĩnh vực khác nhau, bao gồm nhận dạng hình ảnh, xử lý ngôn ngữ tự nhiên, dự đoán, phân loại, gợi ý và rất nhiều ứng dụng khác.

Quá trình xây dựng một mô hình học máy thường bao gồm các bước sau:

1. Tiền xử lý dữ liệu: Chuẩn bị dữ liệu huấn luyện bằng cách làm sạch, xử lý và biến đổi dữ liệu thành định dạng phù hợp để đưa vào mô hình.
2. Lựa chọn mô hình: Chọn một thuật toán hoặc một tập hợp các thuật toán phù hợp với vấn đề và loại dữ liệu cụ thể.
3. Huấn luyện mô hình: Đưa dữ liệu huấn luyện vào mô hình và điều chỉnh các tham số để mô hình có thể học từ dữ liệu và tạo ra các dự đoán chính xác.
4. Đánh giá mô hình: Đánh giá hiệu suất của mô hình bằng cách sử dụng các phương pháp đánh giá phù hợp, như chia dữ liệu thành tập huấn luyện và tập kiểm tra, độ chính xác, độ phân loại, hoặc các phương pháp đánh giá khác.
5. Triển khai mô hình: Đưa mô hình vào ứng dụng thực tế để thực hiện các dự đoán hoặc phân loại cho dữ liệu mới.

Mô hình học máy là một công cụ quan trọng trong lĩnh vực trí tuệ nhân tạo và đóng vai trò quan trọng trong việc giải quyết các bài toán phức tạp và đưa ra dự đoán từ dữ liệu.

## D. Classification

Classification (Phân loại) là một thuật toán trong lĩnh vực học máy, nó nhằm mục đích gán các mẫu dữ liệu vào các em hoặc lớp khác nhau dựa trên các đặc trưng quan sát được. Mục tiêu của phân loại là tìm ra quy luật hoặc mô hình dự đoán để phân loại các mẫu dữ liệu mới vào các em đã biết trước.

Phân loại được ứng dụng rộng rãi trong nhiều lĩnh vực, bao gồm nhận dạng ảnh, phân loại văn bản, phân loại email spam, phát hiện gian lận tín dụng, và nhiều bài toán khác trong việc phân loại và phân đoạn dữ liệu.

### 2.2 CÁC NGHIÊN CỨU LIÊN QUAN

Dưới đây là một số nghiên cứu liên quan đến đề tài "Đào tạo và triển khai mô hình học máy bằng cách sử dụng một EC2 Instance trên AWS":

- "Scalable Machine Learning on AWS using Spark" của S. Karimulla, et al. (2017): Nghiên cứu này tìm hiểu cách sử dụng EC2 Instances trên AWS để xây dựng và triển khai các mô hình học máy có khả năng mở rộng sử dụng Apache Spark. Nghiên cứu tập trung vào việc tối ưu hóa hiệu suất và hiệu quả của việc xử lý dữ liệu lớn và huấn luyện mô hình trên nền tảng AWS.
- "Machine Learning Model Deployment in the Cloud: A Case Study on Amazon EC2" của M. V. S. Barreto, et al. (2020): Nghiên cứu này đề xuất một phương pháp triển khai mô hình học máy trên EC2 Instance của AWS. Nghiên cứu tập trung vào việc xác định các yếu tố ảnh hưởng đến hiệu suất và khả năng mở rộng của mô hình, cũng như tối ưu hóa việc triển khai mô hình trên nền tảng EC2.
- "Cloud-based Machine Learning for Large-Scale Data Analysis" của A. K. Sangaiah, et al. (2018): Nghiên cứu này trình bày một phương pháp sử dụng EC2 Instances trên AWS để huấn luyện và triển khai mô hình học máy cho phân tích dữ liệu quy mô lớn. Nghiên cứu tập trung vào việc tối ưu hóa việc xử lý dữ liệu và huấn luyện mô hình để đạt được hiệu suất cao và thời gian đáp ứng nhanh.
- "Evaluating Performance of Machine Learning Algorithms on AWS EC2 Instances" của A. P. Singh, et al. (2019): Nghiên cứu này so sánh hiệu suất của các thuật toán học máy khác nhau trên EC2 Instances của AWS. Nghiên cứu đánh giá các yếu tố như thời gian huấn luyện, độ chính xác và tài nguyên tiêu thụ để đưa ra sự lựa chọn phù hợp cho việc triển khai mô hình học máy trên EC2.

Các nghiên cứu trên cung cấp cái nhìn tổng quan về việc sử dụng EC2 Instances trên AWS để đào tạo và triển khai mô hình học máy. Tuy nhiên, việc tìm hiểu thêm về các nghiên cứu cụ thể này và các công trình liên quan khác sẽ cung cấp thông tin chi tiết và phong phú hơn về phạm vi và kết quả của các nghiên cứu trong lĩnh vực này.

## CHƯƠNG 3

# PHƯƠNG PHÁP LUẬN

### 3.1 CÁC CÔNG CỤ, CÔNG NGHỆ SỬ DỤNG

#### A. Amazon Web Services ( AWS )

Amazon Web Services (AWS) là một dịch vụ đám mây cung cấp bởi Amazon.com. Nó cung cấp một loạt các dịch vụ đám mây, bao gồm tính toán, lưu trữ, cơ sở dữ liệu, máy học, trí tuệ nhân tạo, Internet of Things (IoT), blockchain và nhiều dịch vụ khác. AWS cho phép các tổ chức và cá nhân xây dựng và triển khai ứng dụng và dịch vụ trên nền tảng đám mây một cách linh hoạt và có khả năng mở rộng.

Amazon Elastic Compute Cloud (EC2): Dịch vụ cung cấp máy ảo đám mây cho việc chạy ứng dụng và công việc tính toán trên nền tảng đám mây của AWS.

AWS cung cấp một số lợi ích như linh hoạt, mở rộng, bảo mật và khả năng tính toán mạnh mẽ. Nó được sử dụng rộng rãi bởi các tổ chức và doanh nghiệp trên toàn cầu để xây dựng và triển khai các ứng dụng và dịch vụ đám mây.



## B. Postman

Postman là một công cụ phát triển ứng dụng API (Application Programming Interface) mạnh mẽ và phổ biến. Nó cung cấp một giao diện đồ họa để sử dụng cho phép người dùng gửi và nhận các yêu cầu HTTP (như GET, POST, PUT, DELETE) đến các API và xem các phản hồi tương ứng.



Dưới đây là một số tính năng và lợi ích chính của Postman:

- **Gửi yêu cầu HTTP:** Postman cho phép bạn tạo và gửi các yêu cầu HTTP đến các API bằng các phương thức như GET, POST, PUT, DELETE và nhiều phương thức khác. Bạn có thể cung cấp các tham số, tiêu đề và thân yêu cầu để tương tác với các API.
- **Quản lý và kiểm thử API:** Postman cho phép bạn tổ chức và quản lý các yêu cầu API trong các bộ sưu tập. Bạn có thể tạo, chỉnh sửa và xóa các yêu cầu, và tổ chức chúng thành các em logic để dễ dàng tìm kiếm và sử dụng lại.
- **Kiểm thử và gỡ lỗi:** Postman cung cấp các tính năng mạnh mẽ để kiểm thử và gỡ lỗi API. Bạn có thể xem các phản hồi từ API, kiểm tra và xác minh dữ liệu trả về, và xử lý lỗi hoặc vấn đề gặp phải trong quá trình tương tác với API.
- **Tích hợp với công cụ khác:** Postman tích hợp tốt với các công cụ phát triển phổ biến khác như Git, Jenkins, Newman và nhiều công cụ khác. Điều này cho phép bạn tích hợp quy trình kiểm thử API vào quy trình phát triển phần mềm tổng thể của bạn.
- **Chia sẻ và hợp tác:** Postman cho phép bạn chia sẻ các bộ sưu tập, yêu cầu và môi trường làm việc với đồng đội. Bạn có thể dễ dàng chia sẻ tệp cấu hình Postman và làm việc cùng nhau trên các yêu cầu API.

## C. WinSCP

WinSCP là một ứng dụng mã nguồn mở và miễn phí được sử dụng để truyền tải tệp tin qua giao thức SFTP (SSH File Transfer Protocol), FTP (File Transfer Protocol), FTPS (FTP over SSL/TLS) và SCP (Secure Copy Protocol). Nó được phát triển cho hệ điều hành Windows và cung cấp giao diện người dùng để sử dụng và trực quan.



Dưới đây là một số tính năng chính của WinSCP:

- Giao diện người dùng để sử dụng: WinSCP cung cấp giao diện kép (dual-pane) cho phép bạn duyệt và quản lý các tệp tin trên máy tính cục bộ và máy chủ từ xa một cách dễ dàng.
- Truyền tải an toàn: WinSCP hỗ trợ các giao thức truyền tải an toàn như SFTP và SCP, giúp bảo mật dữ liệu trong quá trình truyền tải. Nó cũng hỗ trợ FTPS để cung cấp truyền tải qua SSL/TLS.
- Quản lý kết nối và phiên làm việc: WinSCP cho phép lưu trữ thông tin kết nối và thông tin đăng nhập vào các máy chủ từ xa. Bạn có thể lưu các phiên làm việc và sử dụng lại chúng một cách dễ dàng.
- Chức năng đồng bộ hóa: WinSCP cung cấp chức năng đồng bộ hóa tệp tin và thư mục giữa máy tính cục bộ và máy chủ từ xa. Bạn có thể thực hiện đồng bộ hóa một cách tự động hoặc thủ công, đảm bảo rằng các tệp tin trên cả hai bên được cập nhật và đồng bộ.
- Quản lý quyền truy cập: WinSCP cho phép bạn quản lý quyền truy cập trên máy chủ từ xa, bao gồm quyền đọc, ghi và thực thi tệp tin và thư mục. Bạn có thể thay đổi quyền truy cập của tệp tin và thư mục từ giao diện người dùng của WinSCP.

WinSCP là một công cụ hữu ích cho việc truyền tải và quản lý tệp tin qua các giao thức truyền tải khác nhau. Nó được sử dụng rộng rãi trong việc làm việc với máy chủ từ xa và quản lý tệp tin trên môi trường Windows.

#### **D. PuTTY**

PuTTY là một ứng dụng mã nguồn mở và miễn phí được sử dụng để thiết lập kết nối bảo mật SSH (Secure Shell) và Telnet với máy chủ từ xa. Nó được phát triển chủ yếu cho các hệ điều hành Windows, và nó cung cấp giao diện dòng lệnh cho việc truy cập và quản lý máy chủ từ xa.



Dưới đây là một số tính năng chính của PuTTY:

- Truy cập máy chủ từ xa: PuTTY cho phép bạn thiết lập kết nối SSH và Telnet đến các máy chủ từ xa. Bạn có thể truy cập vào hệ thống Linux, Unix và các thiết bị mạng khác để thực hiện các tác vụ quản lý và điều khiển từ xa.
- Bảo mật: PuTTY hỗ trợ các phiên bản của giao thức SSH, bao gồm SSH-1 và SSH-2, để đảm bảo kết nối an toàn giữa máy tính của bạn và máy chủ từ xa. Nó sử dụng mã hóa và xác thực để bảo vệ dữ liệu và thông tin đăng nhập trong quá trình truyền tải.
- Giao diện dòng lệnh: PuTTY cung cấp một giao diện dòng lệnh đơn giản và trực quan. Bạn có thể nhập các lệnh và thực thi chúng trên máy chủ từ xa. Nó hỗ trợ cú pháp màu sắc, xác thực khóa công khai, và các tính năng linh hoạt khác để làm việc hiệu quả với máy chủ từ xa.
- Hỗ trợ Telnet: Bên cạnh SSH, PuTTY cũng hỗ trợ kết nối Telnet. Telnet là một giao thức không bảo mật, nhưng vẫn có thể hữu ích trong một số trường hợp khi không yêu cầu mức độ bảo mật cao.



- Quản lý kết nối: PuTTY cho phép bạn lưu trữ thông tin kết nối và thông tin đăng nhập vào các máy chủ từ xa. Bạn có thể lưu các phiên kết nối và sử dụng lại chúng một cách dễ dàng, giúp tiết kiệm thời gian và nâng cao hiệu suất làm việc.

PuTTY là một công cụ phổ biến và mạnh mẽ cho việc truy cập và quản lý máy chủ từ xa. Với tính năng bảo mật và khả năng tùy chỉnh linh hoạt, nó là một công cụ hữu ích cho các nhà quản trị hệ thống và lập trình viên khi làm việc với các môi trường máy chủ từ xa.

## E. Operating System

- Ubuntu:

Ubuntu là một hệ điều hành mã nguồn mở và miễn phí dựa trên nền tảng Linux. Nó là một trong những phiên bản phổ biến nhất của hệ điều hành Linux và được phát triển và duy trì bởi Canonical Ltd.



Dưới đây là một số điểm nổi bật về Ubuntu:

- Môi trường máy tính để bàn: Ubuntu cung cấp một phiên bản máy tính để bàn với giao diện người dùng đồ họa thân thiện và dễ sử dụng. Giao diện mặc định của Ubuntu là GNOME, nhưng cũng có sẵn các biến thể khác như Kubuntu (với môi trường KDE) và Xubuntu (với môi trường Xfce).
- Hỗ trợ cộng đồng: Ubuntu có một cộng đồng lớn và nhiệt tình của người dùng và nhà phát triển. Người dùng có thể tìm kiếm hỗ trợ từ cộng đồng thông qua diễn đàn, wiki, và các kênh trò chuyện trực tuyến. Cộng đồng Ubuntu cũng đóng góp vào việc phát triển phiên bản và cung cấp các gói phần mềm bổ sung.
- Bảo mật và ổn định: Ubuntu đặt nặng vào bảo mật và cung cấp các bản vá lỗi và bảo mật định kỳ. Nó cũng sử dụng AppArmor và UFW (Uncomplicated Firewall) để giúp bảo vệ hệ thống khỏi các mối đe dọa.

- Phần mềm mã nguồn mở: Ubuntu được xây dựng với sự tập trung vào phần mềm mã nguồn mở. Nó đi kèm với nhiều phần mềm mã nguồn mở phổ biến như LibreOffice (bộ ứng dụng văn phòng), Firefox (trình duyệt web), và GIMP (chương trình chỉnh sửa ảnh).
- Hỗ trợ lâu dài: Ubuntu có chính sách hỗ trợ lâu dài (Long-Term Support - LTS) cho các phiên bản cụ thể. Các phiên bản LTS nhận được hỗ trợ bảo mật và bản vá lỗi trong thời gian dài, giúp người dùng duy trì tính ổn định và an ninh của hệ thống.

Ubuntu là một hệ điều hành linh hoạt và phổ biến, được sử dụng trên máy tính để bàn, máy chủ, và các thiết bị nhúng. Nó cung cấp một môi trường ổn định, dễ sử dụng, và hỗ trợ mạnh mẽ từ cộng đồng người dùng.

## **F. Development Environment**

- Visual Studio Code ( VSCode ):

Visual Studio Code (VSCode) là một trình soạn thảo mã nguồn mở và miễn phí được phát triển bởi Microsoft. Nó hỗ trợ nhiều ngôn ngữ lập trình và cung cấp một loạt tính năng mạnh mẽ để tăng năng suất phát triển phần mềm.



Dưới đây là một số điểm nổi bật về Visual Studio Code:

- Đa nền tảng: VSCode có sẵn cho Windows, macOS và Linux, cho phép người dùng phát triển trên nhiều hệ điều hành khác nhau.
- Mở rộng và tùy chỉnh: VSCode có một hệ thống mở rộng mạnh mẽ cho phép người dùng cài đặt các tiện ích mở rộng (extensions) để mở rộng chức năng và tích hợp với các công cụ và dịch vụ bên thứ ba. Nó cũng cho phép người dùng tùy chỉnh giao diện và các cài đặt khác để phù hợp với nhu cầu cá nhân.
- Hỗ trợ ngôn ngữ đa dạng: VSCode hỗ trợ rất nhiều ngôn ngữ lập trình phổ biến, bao gồm JavaScript, TypeScript, Python, C++, Java, Go, PHP, và nhiều

ngôn ngữ khác. Nó cung cấp các tính năng như kiểm tra lỗi cú pháp, gợi ý mã, định dạng mã tự động, và gỡ lỗi tích hợp.

- Tích hợp hệ thống quản lý phiên bản: VSCode tích hợp tốt với các hệ thống quản lý phiên bản như Git, SVN và Mercurial, giúp người dùng quản lý mã nguồn và tiến trình phát triển dễ dàng hơn.
- Gỡ lỗi và kiểm tra mã: VSCode cung cấp các tính năng gỡ lỗi mạnh mẽ, cho phép người dùng dừng và theo dõi quá trình thực thi mã và tìm ra lỗi. Nó cũng hỗ trợ kiểm tra mã thông qua các phần mở rộng bổ sung như ESLint cho JavaScript và Pylint cho Python.
- Hỗ trợ cộng đồng: VSCode có một cộng đồng sôi nổi và tích cực, cung cấp các tiện ích mở rộng, chủ đề, và tài nguyên hữu ích. Người dùng có thể tìm kiếm thông qua Marketplace để tìm và cài đặt các tiện ích mở rộng và chủ đề phù hợp với nhu cầu của mình.

Visual Studio Code đã trở thành một trình soạn thảo mã nguồn phổ biến và được ưa chuộng trong cộng đồng phát triển phần mềm nhờ vào tính linh hoạt, tính mở rộng và khả năng tùy chỉnh của nó.

## **G. Phát triển ứng dụng Web**

- Streamlit:

Streamlit là một framework phát triển ứng dụng web tương tác và trực quan cho việc phân tích dữ liệu và máy học (data science và machine learning) bằng Python. Nó được thiết kế để giúp người dùng nhanh chóng xây dựng các ứng dụng web chỉ với ít công sức và không cần kiến thức sâu về phát triển web.



Một số đặc điểm chính của Streamlit bao gồm:

- Dễ sử dụng: Streamlit cung cấp một API đơn giản và dễ hiểu, giúp người dùng dễ dàng hiển thị dữ liệu, tạo biểu đồ và tương tác với người dùng thông qua các câu lệnh Python.
- Tự động làm mới: Khi bạn thay đổi mã nguồn của ứng dụng Streamlit, trình duyệt sẽ tự động làm mới và hiển thị các thay đổi mà không cần khởi động lại ứng dụng.
- Hỗ trợ đa loại dữ liệu: Streamlit hỗ trợ nhiều loại dữ liệu như văn bản, hình ảnh, âm thanh và video. Bạn có thể dễ dàng hiển thị chúng trong ứng dụng của mình.
- Tích hợp dễ dàng: Streamlit tích hợp tốt với các thư viện phổ biến trong ngành phân tích dữ liệu và máy học như Pandas, Matplotlib, Plotly và Scikit-learn. Điều này giúp bạn tận dụng lại mã nguồn hiện có và tích hợp các tính năng mạnh mẽ vào ứng dụng của mình.
- Tích hợp triển khai: Streamlit cung cấp các công cụ để triển khai ứng dụng trực tiếp từ máy tính cá nhân lên các dịch vụ đám mây như Heroku, hoặc tạo thành các ứng dụng độc lập có thể chạy trên nhiều nền tảng.

Streamlit đã trở thành một công cụ phổ biến trong cộng đồng phân tích dữ liệu và máy học nhờ vào tính đơn giản và khả năng tạo ra các ứng dụng web tương tác nhanh chóng.

- Flask:

Flask là một micro-framework web mã nguồn mở được viết bằng Python. Được phát triển bởi Armin Ronacher, Flask được thiết kế nhằm đơn giản hóa việc xây dựng ứng dụng web nhỏ và trung bình. Đặc trưng của Flask là linh hoạt và dễ dùng, cho phép người dùng tùy chỉnh và mở rộng theo nhu cầu cụ thể.



Dưới đây là một số điểm nổi bật của Flask:

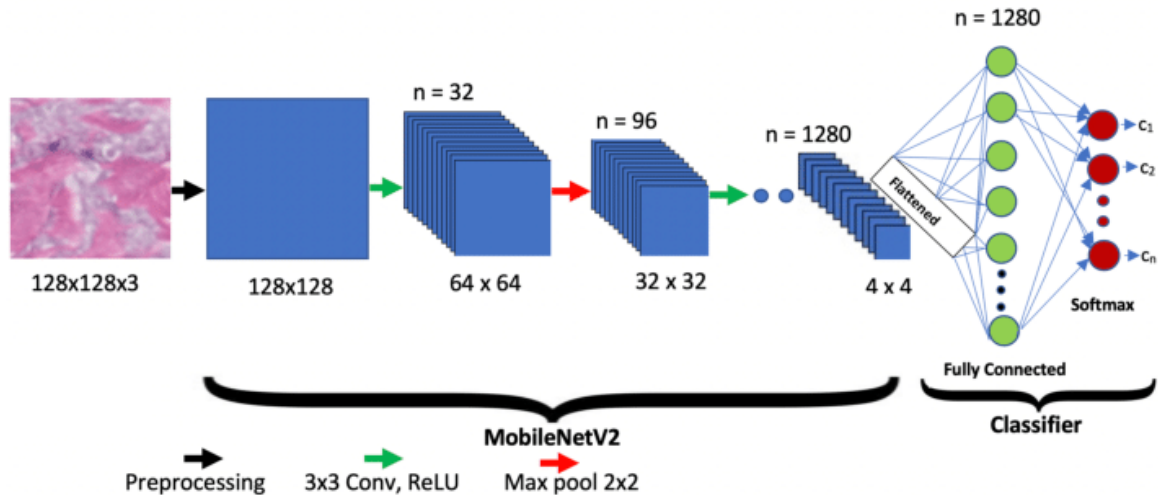
- **Micro-framework:** Flask được gọi là micro-framework vì nó chỉ cung cấp các thành phần cơ bản để xây dựng các ứng dụng web. Tuy nhiên, Flask cho phép người dùng linh hoạt mở rộng và tích hợp các thư viện và công cụ mở rộng để đáp ứng nhu cầu cụ thể.
- **Đơn giản và dễ sử dụng:** Flask có một cú pháp đơn giản và dễ hiểu, giúp người dùng nhanh chóng khởi tạo ứng dụng web và xây dựng các tuyến đường (routes) và xử lý yêu cầu (request handling) đơn giản.
- **Tích hợp Jinja2:** Flask tích hợp sẵn Jinja2, một hệ thống mẫu mạnh mẽ cho phép người dùng tạo giao diện động và tái sử dụng mã HTML.
- **Hỗ trợ RESTful:** Flask hỗ trợ xây dựng các ứng dụng RESTful, cho phép người dùng tạo các API cho việc truy xuất và thay đổi dữ liệu.
- **Mở rộng và tùy chỉnh:** Flask có một cộng đồng rộng lớn và nhiều thư viện mở rộng (extensions) để hỗ trợ các tính năng bổ sung như xử lý đăng nhập, xác thực, gửi email, và quản lý phiên bản.
- **Triển khai dễ dàng:** Flask cho phép người dùng triển khai ứng dụng web trên nhiều nền tảng, từ máy tính cá nhân đến các dịch vụ đám mây như Heroku và AWS.

Flask là một lựa chọn phổ biến cho việc xây dựng các ứng dụng web nhỏ và trung bình, đặc biệt là trong việc xây dựng API và ứng dụng web đơn giản.

## **H. Machine Learning Model**

- **MobileNetV2:**

MobileNetV2 là một kiến trúc mạng neural convolutional (CNN) được phát triển bởi em nghiên cứu Google Brain vào năm 2018. Được xây dựng trên cơ sở của MobileNetV1, MobileNetV2 cải thiện hiệu suất và hiệu quả tính toán của mô hình.



MobileNetV2 nhằm mục tiêu cung cấp một mô hình CNN nhẹ nhàng và hiệu quả để triển khai trên các thiết bị di động và hạn chế tài nguyên tính toán. Nó sử dụng một số kỹ thuật chính để đạt được mục tiêu này:

- **Depthwise Separable Convolution:** MobileNetV2 sử dụng phép tích chập chia để chinh phục số lượng tham số và tính toán. Thay vì áp dụng một phép tích chập thông thường với các kernel lớn, mô hình sử dụng phép tích chập theo chiều sâu (depthwise convolution) và phép tích chập 1x1 (pointwise convolution) để giảm tỷ lệ tham số và tính toán.
- **Bottleneck Residual Block:** MobileNetV2 sử dụng các khối bottleneck residual để tăng cường hiệu suất mô hình. Khối này bao gồm các lớp tích chập 1x1 và 3x3 để giảm kích thước đầu vào và tạo ra các biểu diễn trung gian có chiều sâu lớn hơn.
- **Linear Bottlenecks:** MobileNetV2 sử dụng hàm kích hoạt linearity trong các khối bottleneck để giảm các ảnh hưởng không mong muốn đến hiệu năng của mô hình.

MobileNetV2 đã chứng tỏ hiệu quả và độ chính xác cao trong nhiều nhiệm vụ thị giác máy tính như phân loại ảnh và nhận dạng đối tượng trên các thiết bị di động có tài nguyên hạn chế. Nó đã trở thành một trong những mô hình phổ biến được sử dụng trong lĩnh vực trí tuệ nhân tạo và thị giác máy tính.

### 3.2 THIẾT KẾ KIẾN TRÚC ĐỀ TÀI




- Đào tạo mô hình và lưu model dưới dạng file .h5: Sử dụng Jupyter Notebook để đào tạo mô hình học máy. Sau khi hoàn thành quá trình đào tạo, lưu mô hình dưới dạng file .h5, sử dụng thư viện như Keras, TensorFlow.
- Truyền dữ liệu từ máy cục bộ sang EC2 bằng WinSCP: Sử dụng phần mềm WinSCP để truyền dữ liệu từ máy tính cục bộ sang EC2 instance trên AWS. Kết nối với EC2 instance thông qua giao thức SSH và sao chép dữ liệu từ máy tính cục bộ sang EC2.
- Chạy và triển khai mô hình trên EC2: Trên EC2 instance, cài đặt các môi trường và thư viện cần thiết cho việc chạy mô hình học máy. Sau đó, tải file .h5 chứa mô hình đã được lưu trữ từ bước trước và chạy mô hình trên EC2.
- Sử dụng Flask và Streamlit để tạo giao diện web tương tác: Sử dụng Flask hoặc Streamlit, hai framework phổ biến cho phát triển ứng dụng web Python, để tạo giao diện web tương tác với người dùng. Trang web cho phép người dùng nhập dữ liệu và gửi nó đến EC2 để xử lý.
- Nhập dữ liệu từ người dùng và xử lý trên EC2: Khi người dùng nhập dữ liệu vào trang web, dữ liệu sẽ được gửi từ trình duyệt của người dùng đến EC2 instance. Tại đó, ứng dụng web sử dụng mô hình đã triển khai để xử lý dữ liệu và phân loại nó.
- Gửi kết quả phân loại trở lại trang web: Sau khi dữ liệu được xử lý và phân loại trên EC2 instance, kết quả phân loại sẽ được gửi trở lại trang web tương tác. Người dùng có thể xem kết quả trên trình duyệt của mình thông qua giao diện web đã được tạo ra bằng Flask hoặc Streamlit.

### 3.3 CÁC BƯỚC THỰC HIỆN ĐỀ TÀI

#### **Bước 1:** Đào tạo mô hình và lưu model

Để huấn luyện mô hình nhận diện hình ảnh trái cây và rau củ từ tập dữ liệu "Fruit and Vegetable Image Recognition" gồm 36 lớp, mỗi lớp có khoảng 100 hình ảnh, tổng cộng hơn 2800 hình ảnh.

 KRITIK SETH · UPDATED 2 YEARS AGO


273

New Notebook

Download (2 GB)

## Fruits and Vegetables Image Recognition Dataset

Fruit and Vegetable Images for Object Recognition



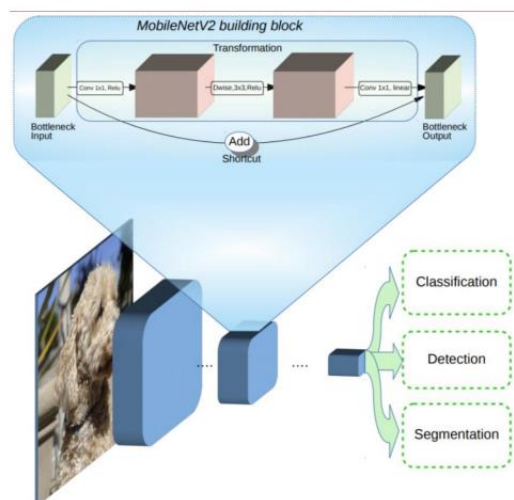
```
-- Training set --

Number of pictures: 2855

Number of different labels: 36

Labels: ['capsicum' 'pomegranate' 'eggplant' 'grapes' 'garlic' 'mango' 'soy beans'
'orange' 'paprika' 'pear' 'onion' 'peas' 'jalepeno' 'apple' 'cucumber'
'cabbage' 'tomato' 'banana' 'sweetpotato' 'spinach' 'chilli pepper'
'watermelon' 'turnip' 'lettuce' 'sweetcorn' 'corn' 'ginger' 'carrot'
'bell pepper' 'cauliflower' 'pineapple' 'kiwi' 'lemon' 'potato'
'beetroot' 'raddish']
```

#### Sử dụng kiến trúc MobilenetV2





## Bước 2: Launch Instance EC2

Đặt tên và chọn AMI: Ubuntu

**Name and tags** [Info](#)

Name

api\_instance

Add additional tags

**▼ Application and OS Images (Amazon Machine Image)** [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Q

Search our full catalog including 1000s of application and OS images

Recents

Quick Start

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Q

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

**Amazon Machine Image (AMI)**

Ubuntu Server 20.04 LTS (HVM), SSD Volume Type

Free tier eligible ▼

ami-06aa3f7caf3a30282 (64-bit (x86)) / ami-0a75bd84854bc95c9 (64-bit (Arm))

Virtualization: hvm   ENA enabled: true   Root device type: ebs

**Description**

Canonical, Ubuntu, 20.04 LTS, amd64 focal image build on 2023-10-25

Architecture

AMI ID

64-bit (x86) ▼

ami-06aa3f7caf3a30282

Verified provider

25

Chọn Instance type: t2.large

▼ **Instance type** [Info](#) | [Get advice](#)

Instance type

**t2.large**

Family: t2 2 vCPU 8 GiB Memory Current generation: true

On-Demand Windows base pricing: 0.1208 USD per Hour

On-Demand RHEL base pricing: 0.1528 USD per Hour

On-Demand SUSE base pricing: 0.1928 USD per Hour

On-Demand Linux base pricing: 0.0928 USD per Hour

☐ All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

Tạo key pair mới có tên trùng với tên instance

Chọn key pair type: RSA và Private key file format: .ppk

**Create key pair** ✕

Key pair name

Key pairs allow you to connect to your instance securely.

⚠ Key pair already exists.

The name can include upto 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ **RSA**

RSA encrypted private and public key pair

☐ **ED25519**

ED25519 encrypted private and public key pair

Private key file format

☐ **.pem**

For use with OpenSSH

☒ **.ppk**

For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#)

[Cancel](#) [Create key pair](#)

Trong mục cài đặt network cho phép HTTP/ HTTPS traffic from the internet

▼ Network settings Info

Edit

Network Info

vpc-0a94d30f8739b1c16

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

We'll create a new security group called 'launch-wizard-2' with the following rules:

☒ Allow SSH traffic from

Helps you connect to your instance

Anywhere  
0.0.0.0/0

☒ Allow HTTPS traffic from the internet

To set up an endpoint, for example when creating a web server

☒ Allow HTTP traffic from the internet

To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

×

Thêm 1 security group có port range trùng với web muốn phát triển

▼ Security group rule 3 (TCP, 5000, 0.0.0.0/0)

Remove

Type Info

Protocol Info

Port range Info

Custom TCP ▼

TCP

5000

Source type Info

Source Info

Description - optional Info

Anywhere ▼

Q Add CIDR, prefix list or security g

0.0.0.0/0 ✕

e.g. SSH for admin desktop

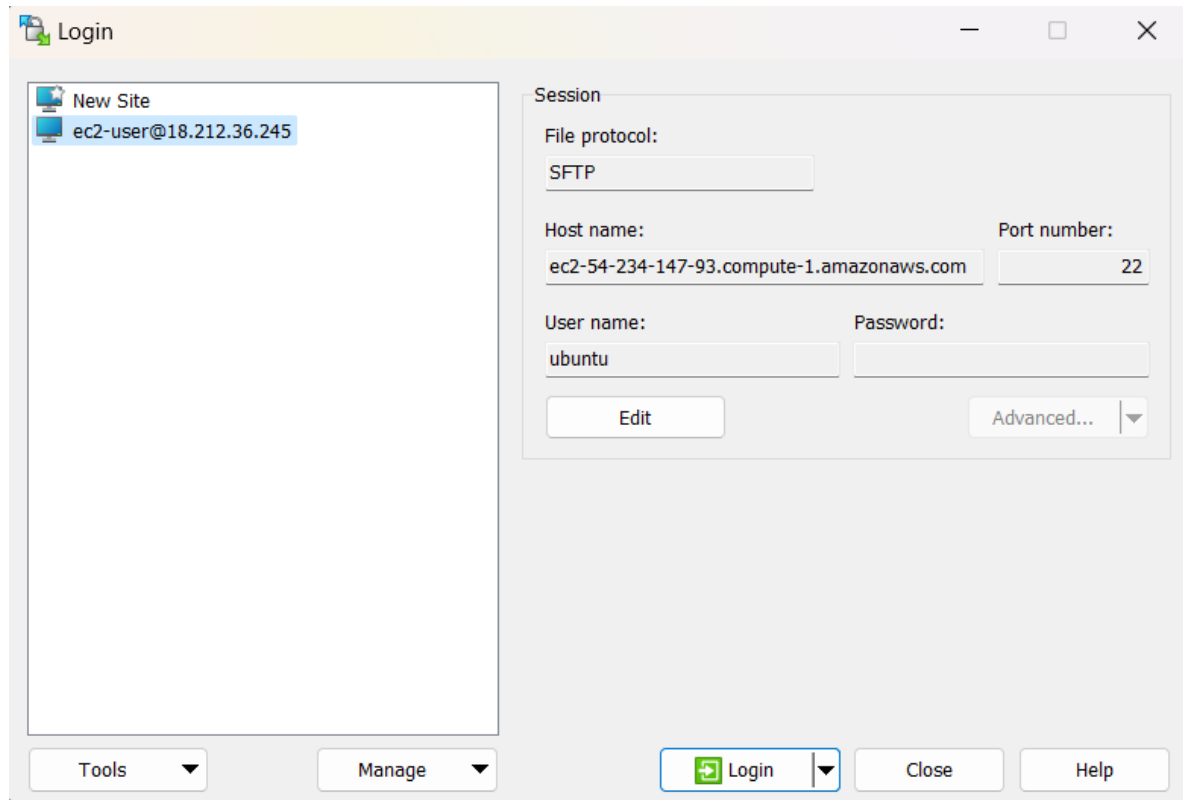
⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

×

*Bắt đầu Launch instance và Connect khi instance state = Running*

### **Bước 3:** Transfer files from local machine to EC2 use WinSCP

- Thiết lập WinSCP để Login tới EC2



- Host name tương ứng với Public IPv4 DNS của instance

Public IPv4 address 54.234.147.93 <a href="#">open address</a>	Private IPv4 addresses 172.31.87.108
Instance state Running	Public IPv4 DNS ec2-54-234-147-93.compute-1.amazonaws.com <a href="#">open address</a>

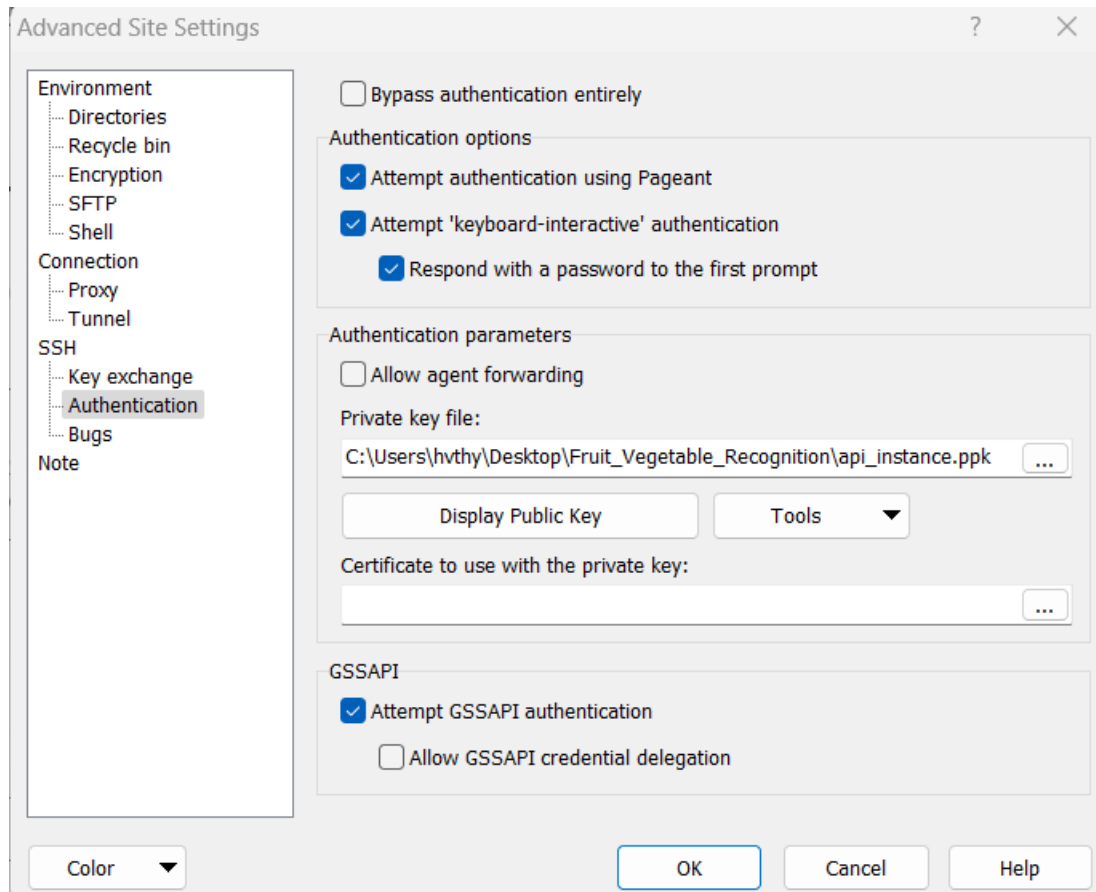
- User name tương ứng với User name của EC2 Instance Connect

Public IP address  
54.234.147.93

User name  
Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.

ubuntu

- Thêm Private key file  
*Chọn Advanced Site Settings -> SSH -> Authentication -> Private key file*

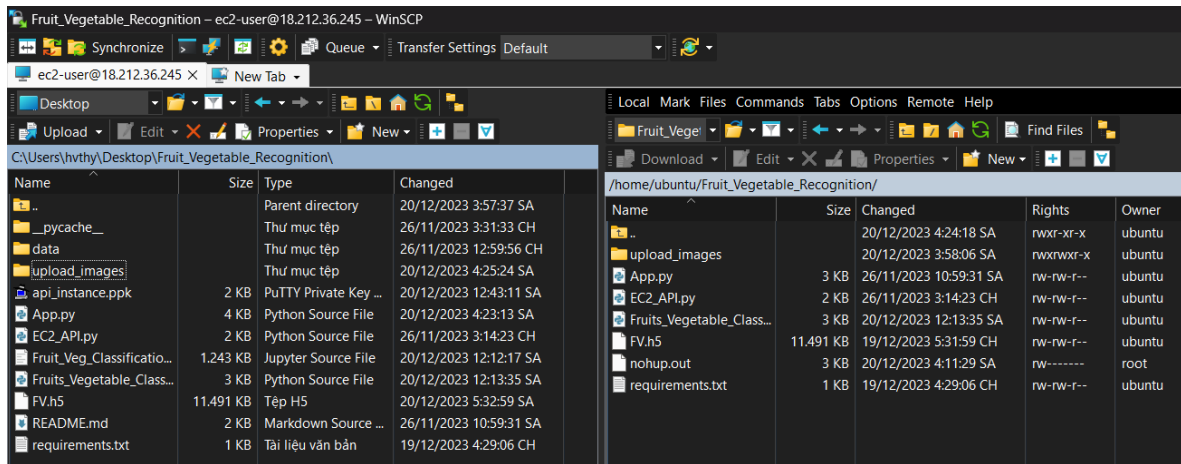


*Private key file là file đã download khi tạo key pair mới lúc launch instance.*

- Login để kết nối tới EC2

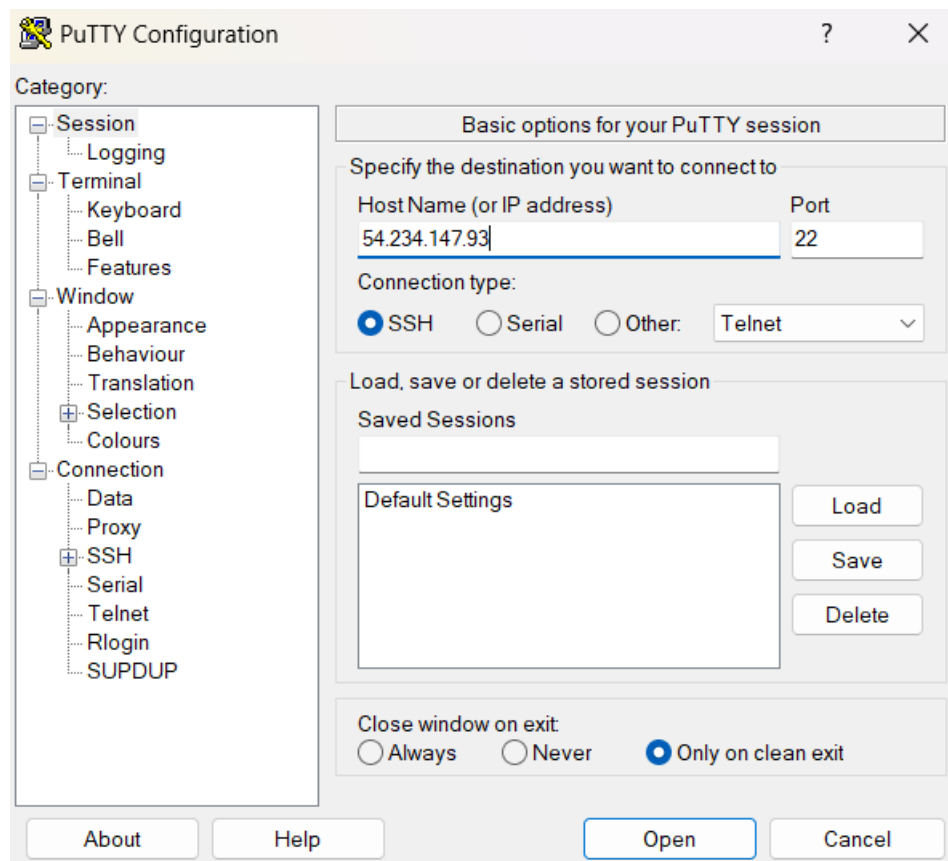
```
ec2-user@18.212.36.245
Searching for host...
Connecting to host...
Authenticating...
Using username "ubuntu".
Authenticating with public key "api_instance".
Authenticated.
Starting the session...
Reading remote directory...
```

- Kéo thả để chuyển dữ liệu cần thiết sang EC2



#### Bước 4: Use PuTTY to connect to EC2 and Deloy model

- Thiết lập PuTTY để Connect tới EC2



- Host name tương ứng với Public IP address của EC2 Instance Connect

EC2 Instance Connect | Session Manager | SSH client | EC2 serial console

Instance ID  
i-05e9d29db583bf39f (api\_instance)

Connection Type

☒ Connect using EC2 Instance Connect  
Connect using the EC2 Instance Connect browser-based client, with a public IPv4 address.

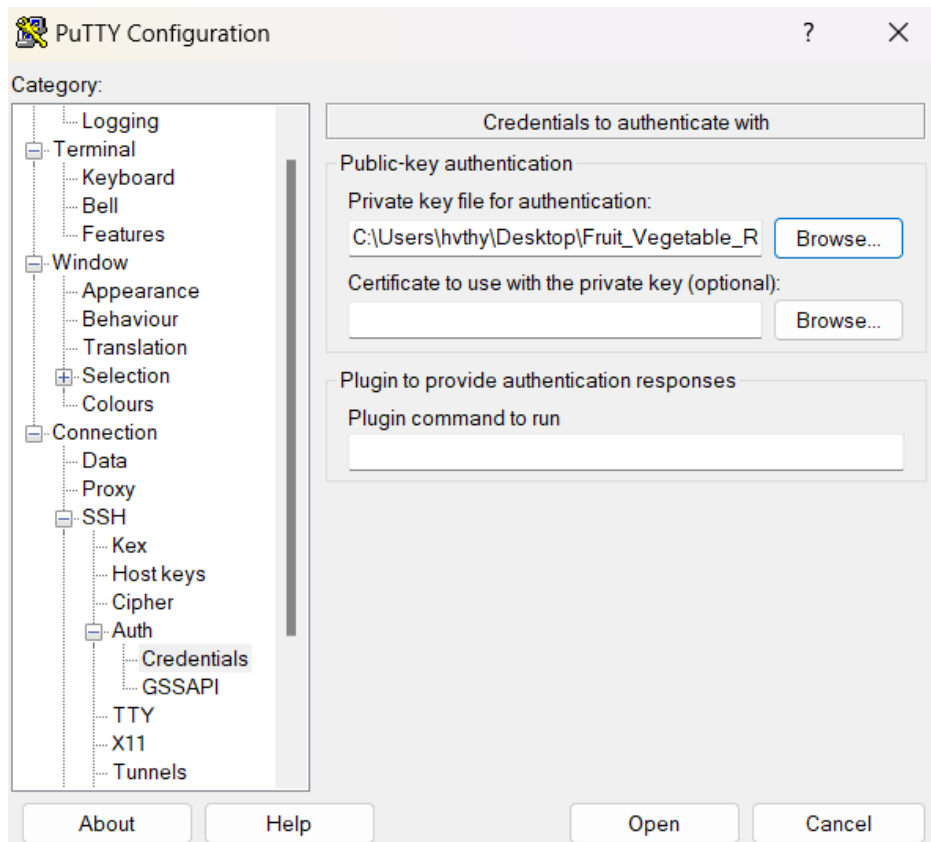
☐ Connect using EC2 Instance Connect Endpoint  
Connect using the EC2 Instance Connect browser-based client, with a private IPv4 address and a VPC endpoint.

Public IP address  
54.234.147.93

User name  
Enter the user name defined in the AMI used to launch the instance. If you didn't define a custom user name, use the default user name, ubuntu.  
ubuntu

**Note:** In most cases, the default user name, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

- Thêm Private key file  
Chọn Connection -> SSH -> Auth -> Credentials -> Private key file



Private key file là file đã download khi tạo key pair mới lúc launch instance.

- Kết nối EC2 và Deloy Model
- Login as: ubuntu ( User name )

```

ubuntu@ip-172-31-87-108: ~
login as: ubuntu
Authenticating with public key "api_instance"
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1048-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

System information as of Tue Dec 19 23:07:15 UTC 2023

System load:  0.0              Processes:            125
Usage of /:   57.6% of 7.57GB   Users logged in:     1
Memory usage: 9%              IPv4 address for eth0: 172.31.87.108
Swap usage:   0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

  https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

52 updates can be applied immediately.
40 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '22.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Tue Dec 19 21:21:42 2023 from 1.53.41.210
ubuntu@ip-172-31-87-108:~$

```

- Chuyển đổi sang tài khoản người dùng root (quyền quản trị viên) trong hệ thống
- Cập nhật danh sách các gói phần mềm trong hệ thống từ các kho lưu trữ chính thức.

```

Last login: Tue Dec 19 21:21:42 2023 from 1.53.41.210
ubuntu@ip-172-31-87-108:~$ sudo su
root@ip-172-31-87-108:/home/ubuntu# apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu focal-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu focal-security InRelease
Fetched 114 kB in 0s (256 kB/s)
Reading package lists... Done

```



- Kiểm tra version của python3

```
root@ip-172-31-87-108:/home/ubuntu# python3
Python 3.8.10 (default, Nov 22 2023, 10:22:35)
[GCC 9.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
root@ip-172-31-87-108:/home/ubuntu#
```

- Kiểm tra các thư mục hiện có, truy cập vào thư mục

```
root@ip-172-31-87-108:/home/ubuntu# ls
Fruit_Vegetable_Recognition
root@ip-172-31-87-108:/home/ubuntu# cd Fruit_Vegetable_Recognition
root@ip-172-31-87-108:/home/ubuntu/Fruit_Vegetable_Recognition#
```

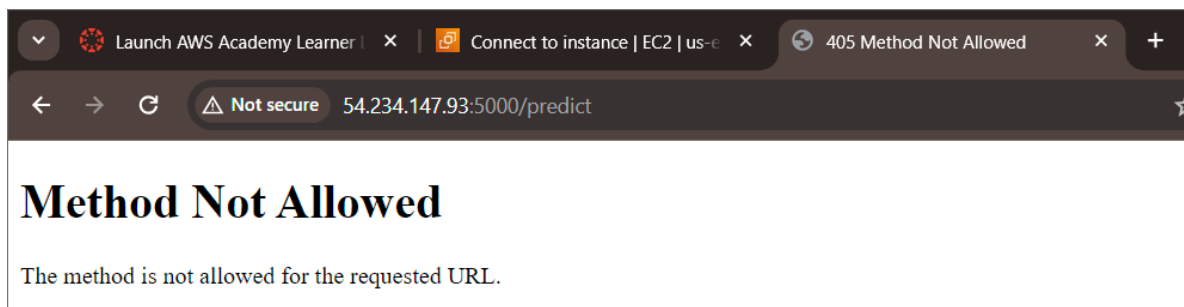
- Cài đặt thư viện

```
root@ip-172-31-87-108:/home/ubuntu/Fruit_Vegetable_Recognition# python3 -m pip install -r requirements.txt
Requirement already satisfied: requests in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (2.22.0)
Requirement already satisfied: BeautifulSoup4 in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 2)) (4.12.2)
Requirement already satisfied: streamlit in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 3)) (1.29.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 4)) (1.24.3)
Requirement already satisfied: Pillow in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 5)) (10.1.0)
Requirement already satisfied: flask in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 6)) (3.0.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 7)) (3.7.4)
Requirement already satisfied: keras in /usr/local/lib/python3.8/dist-packages (from -r requirements.txt (line 8)) (2.13.1)
```

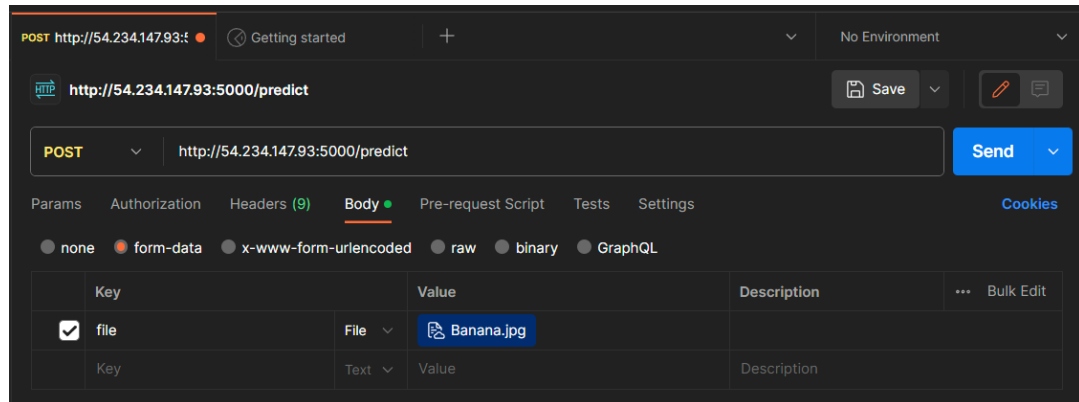
- Chạy chương trình Deploy Model

```
root@ip-172-31-87-108:/home/ubuntu/Fruit_Vegetable_Recognition# python3 EC2_API.py
2023-12-19 23:20:34.817589: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine, GPU will not be used.
2023-12-19 23:20:34.873585: I tensorflow/tsl/cuda/cudart_stub.cc:28] Could not find cuda drivers on your machine, GPU will not be used.
2023-12-19 23:20:34.874201: I tensorflow/core/platform/cpu_feature_guard.cc:182] This TensorFlow binary is optimized to use available CPU
To enable the following instructions: AVX2 FMA, in other operations, rebuild TensorFlow with the appropriate compiler flags.
2023-12-19 23:20:35.767974: W tensorflow/compiler/tf2tensorrt/utils/py_utils.cc:38] TF-TRT Warning: Could not find TensorRT
* Serving Flask app 'EC2_API'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.31.87.108:5000
Press CTRL+C to quit
```

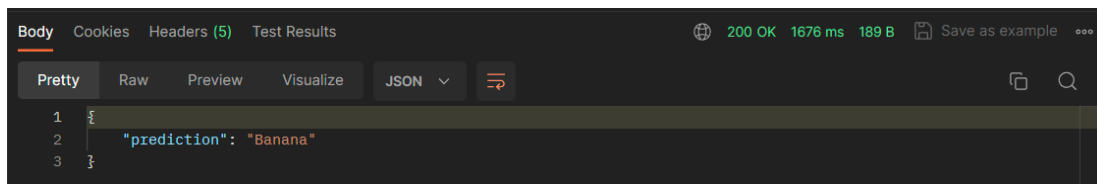
- Truy cập Public IP address của EC2 instance connect để Deploy Model



- Copy url vào Postman để Deloy Model



*Chọn POST để upload data vào model.*



*Kết quả cho prediction đúng với data đã upload vào model*

### **Bước 5:** Deloy on Streamlit

- Copy url vào code để triển khai chương trình lên Web

```
def run():
    st.title("Fruits 🍌 -Vegetable 🍆 Classification")
    img_file = st.file_uploader("Choose an Image", type=["jpg", "png"])
    if img_file is not None:
        img = Image.open(img_file).resize((250, 250))
        st.image(img, use_column_width=False)
        save_image_path = './upload_images/' + img_file.name
        with open(save_image_path, "wb") as f:
            f.write(img_file.getbuffer())

    if img_file is not None:
        url = 'http://54.234.147.93:5000/predict'
        form_data = {'file': open(save_image_path, 'rb')}
        resp = requests.post(url, files=form_data)
        resp_dict = resp.json()
        result = resp_dict['prediction']
        if result in vegetables:
            st.info('**Category : Vegetables**')
        else:
            st.info('**Category : Fruit**')
        st.success('**Predicted : " + result + "**')
        cal = fetch_calories(result)
        if cal:
            st.warning('*** + cal + '(100 grams)***')
```

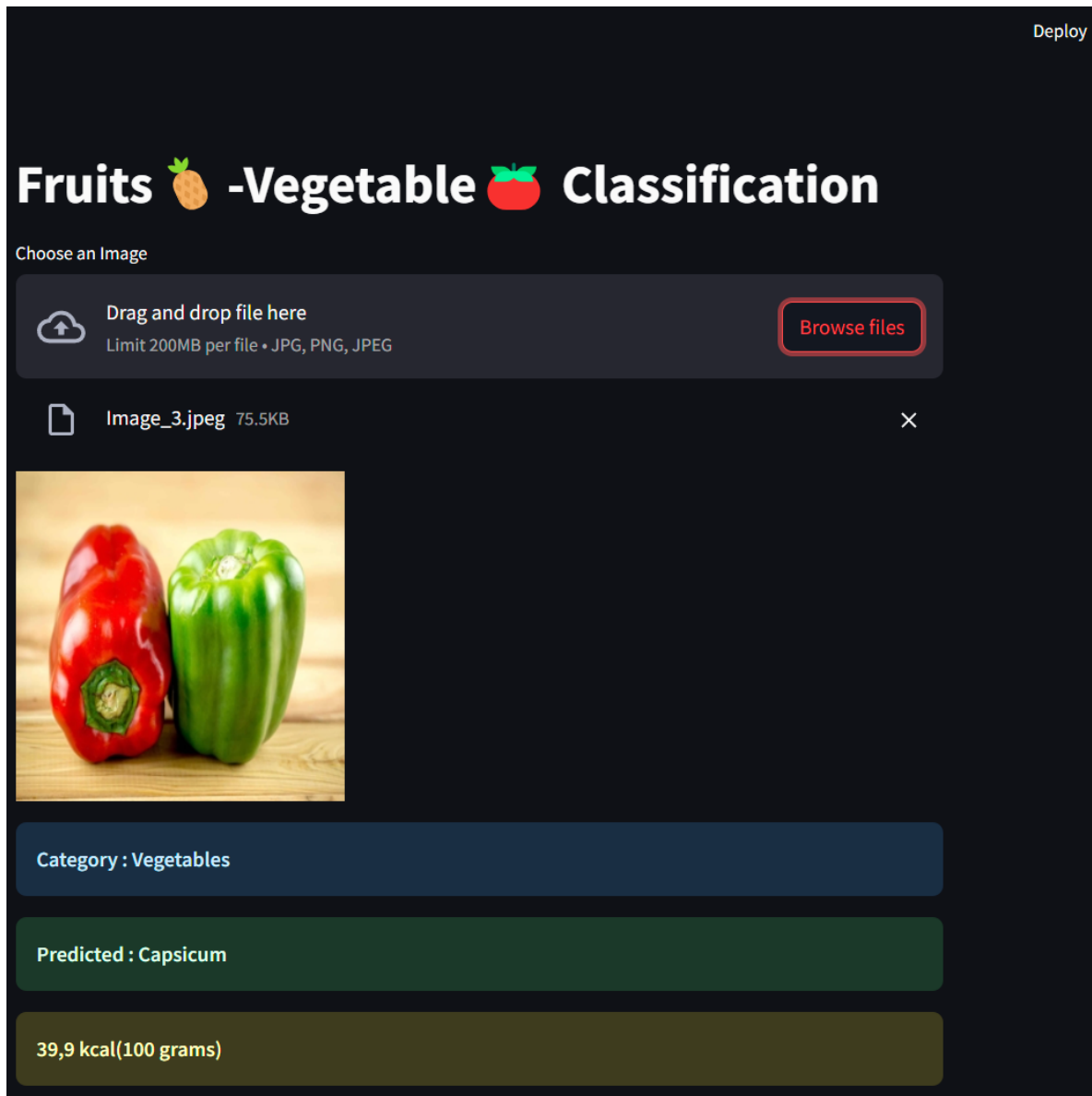
- Chạy chương trình bằng command

```
(tf) C:\Users\hvthy\Desktop\Fruit_Vegetable_Recognition>streamlit run App.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.0.104:8501
```

- Triển khai chương trình trên Streamlit



## CHƯƠNG 4

# QUÁ TRÌNH PHÁT TRIỂN

### 4.1 NHỮNG THÁCH THỨC VÀ GIẢI PHÁP

#### A. Lựa chọn Amazon Machine Image (AMI)

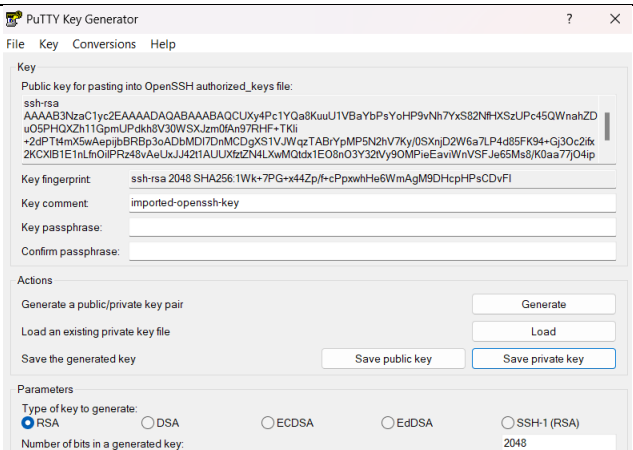
<i>AMI</i>	<i>Thách thức</i>	<i>Giải pháp</i>
Windows	<ul style="list-style-type: none"><li>- Sử dụng remote desktop để kết nối với EC2 khiến cho các thao tác phức tạp hơn</li><li>- Máy ảo EC2 có hiện tượng lag, chậm, đơ rất khó để thực hiện lưu loát các hành động</li><li>- Truyền file từ local machine sang EC2 rất chậm</li></ul>	Sau nhiều lần thử thì em rút ra được một giải pháp chung nhất là sử dụng AMI Ubuntu.
Amazon Linux (Centos)	<ul style="list-style-type: none"><li>- Python mặc định là 2.7. Rất khó để sử dụng thư viện Tensorflow</li><li>- Khi tải Python version mới sẽ làm tốn khá nhiều space. Không thể tải thêm các thư viện bổ trợ khác</li></ul>	Do Ubuntu hỗ trợ python 3.8 phù hợp để Deploy Machine Learning model cũng như các thao tác trên Ubuntu cũng rất nhanh chóng và tiện ích.

#### B. Instance type

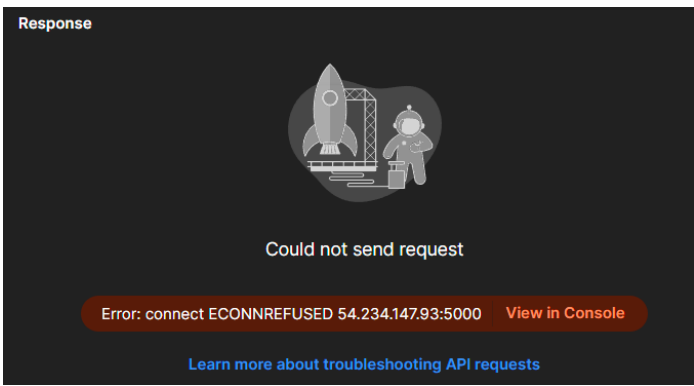
<i>Instance type</i>	<i>Thách thức</i>	<i>Giải pháp</i>
t2.micro	<ul style="list-style-type: none"><li>- Chạy chậm: 1 vCPU</li><li>- Không đủ dung lượng: 1GB</li></ul>	Sử dụng t2.large ( 2 vCPU, 8GB ) mặc dù chi phí đắt hơn nhưng cấu hình của nó phù hợp với việc Deploy Machine Learning model

#### C. Private key file format

<i>Private key file format</i>	<i>Thách thức</i>	<i>Giải pháp</i>
.pem	<ul style="list-style-type: none"><li>- Một số phiên bản cũ của</li></ul>	Sử dụng PuTTYgen để chuyển đổi file từ .pem thành .ppk

	<p>EC2 không có quyền lựa chọn file format</p> <ul style="list-style-type: none"> <li>- Không hỗ trợ trong WinSCP và PuTTY</li> </ul>	
--	---	--

#### D. Khi đóng PuTTY thì không thể Deloy Model

AMI	Thách thức	Giải pháp
Postman		Sử dụng nohup để chương trình có thể chạy trong nền
Streamlit	<pre>ConnectionError: HTTPConnectionPool(host='54.234.147.93', port=5000): Max retries exceeded with url: /predict (Caused by NewConnectionError('&lt;urllib3.connection.HTTPConnection object at 0x00000230341AF850&gt;: Failed to establish a new connection: [WinError 10061] No connection could be made because the target machine actively refused it'))</pre>	

#### E. Chạy chương trình bị lỗi port

Mô tả	Giải pháp
Sau khi cho chương trình chạy trong nền thì chạy lại Deloy model bị lỗi port	<p>Xóa chạy trong nền</p> <pre>ubuntu@ip-172-31-87-108:~\$ sudo su root@ip-172-31-87-108:/home/ubuntu# ps -ef UID          PID    PPID  C  STIME TTY          TIME CMD root           1        0  0   Dec19 ?           00:00:04 /sbin/init root           2        0  0   Dec19 ?           00:00:00 [kthreadd]</pre>

	root	9394	9387	1	00:00	?	00:00:04	python3 EC2_API.py
	root	9408	2	0	00:00	?	00:00:00	[kworker/0:2-cgroup_destroy]
	root	9414	696	0	00:06	?	00:00:00	sshd: ubuntu [priv]
	ubuntu	9487	9414	0	00:06	?	00:00:00	sshd: ubuntu@pts/1
	ubuntu	9488	9487	0	00:06	pts/1	00:00:00	-bash
	root	9497	9488	0	00:06	pts/1	00:00:00	sudo su
	root	9498	9497	0	00:06	pts/1	00:00:00	su
	root	9499	9498	0	00:06	pts/1	00:00:00	bash
	root	9506	9499	0	00:06	pts/1	00:00:00	ps -ef
	root@ip-172-31-87-108:/home/ubuntu# kill 9387							

## 4.2 QUẢN LÝ PHIÊN BẢN VÀ HỢP TÁC

Trong quá trình thực hiện dự án, em đã sử dụng công cụ quản lý phiên bản Git và dịch vụ lưu trữ mã nguồn GitHub. Dưới đây là quá trình quản lý mã nguồn và phiên bản:

1. *Tạo repository trên GitHub*: Em đã tạo một repository trên GitHub để lưu trữ mã nguồn dự án. Điều này cho phép các thành viên trong em làm việc cùng nhau trên cùng một mã nguồn và theo dõi lịch sử thay đổi.

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

*Required fields are marked with an asterisk (\*).*

Owner \*

xiisii

Repository name \*

Cloud-AWS-Project

Cloud-AWS-Project is available.

Great repository names are short and memorable. Need inspiration? How about [effective-parakeet](#) ?

Description (optional)

MACHINE LEARNING MODEL TRAINING AND DEPLOYMENT USING AN AWS EC2 INSTANCE

☒
Public
Anyone on the internet can see this repository. You choose who can commit.

☐
Private
You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

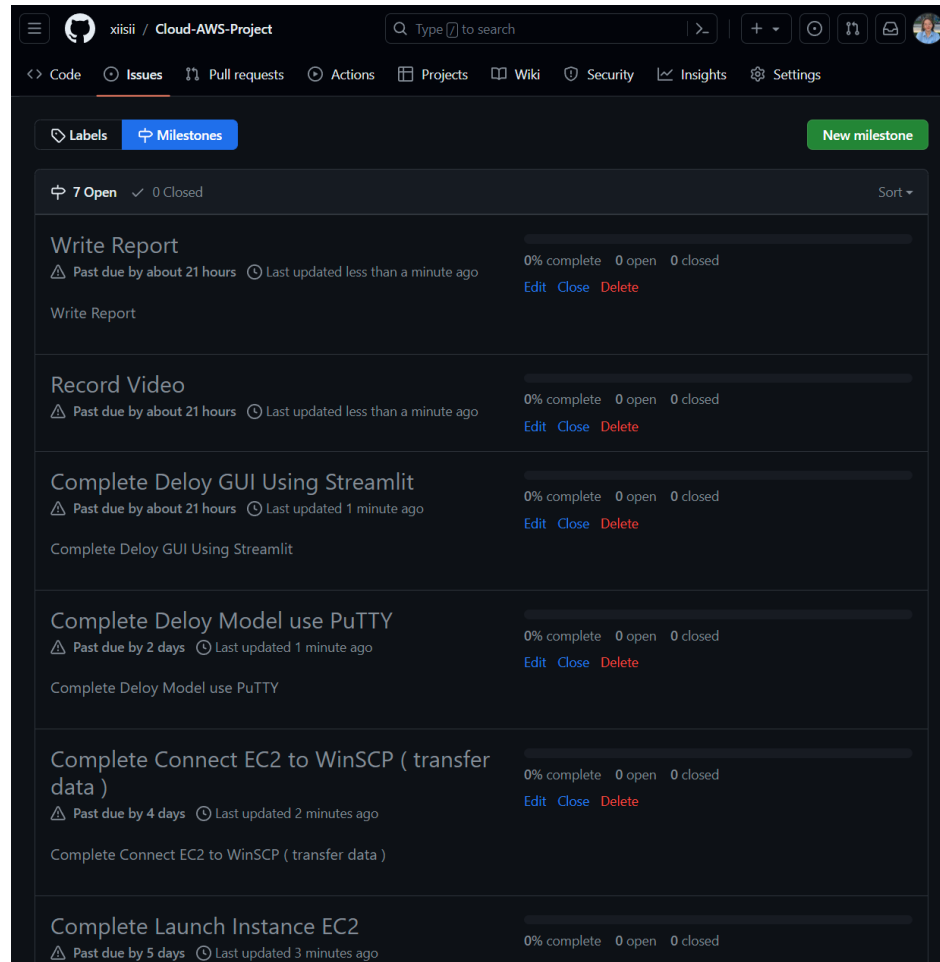
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

Create repository

2. *Tạo các Sprint:* Sau khi đã thiết lập repository, tạo các sprint là một cách hiệu quả để quản lý tính năng và tiến độ dự án.



3. *Branch và merge:* Khi làm việc trên dự án, em đã tạo các nhánh (branch) riêng để phát triển tính năng hoặc sửa đổi mã nguồn mà không ảnh hưởng đến nhánh chính (master). Khi tính năng hoặc sửa đổi hoàn thành, em đã thực hiện quá trình merge để kết hợp các thay đổi vào nhánh chính.

Qua quá trình này, em đã có thể quản lý mã nguồn và phiên bản dự án một cách hiệu quả thông qua Git và GitHub, đồng thời cho phép làm việc đồng thời và theo dõi lịch sử thay đổi của dự án.

# CHƯƠNG 5

## KẾT QUẢ VÀ ĐÁNH GIÁ

### 5.1 KIỂM TRA CHỨC NĂNG

Để kiểm tra chức năng của mô hình học máy được đào tạo và triển khai bằng cách sử dụng EC2 Instance trên AWS, bạn có thể thực hiện các bước sau:

- **Chuẩn bị dữ liệu kiểm tra:** Chuẩn bị một tập dữ liệu kiểm tra đại diện cho loại dữ liệu mà mô hình của bạn sẽ gặp phải trong các tình huống thực tế. Đảm bảo tập dữ liệu kiểm tra này là khác biệt so với tập dữ liệu huấn luyện và tập dữ liệu xác thực để đánh giá khả năng tổng quát hóa của mô hình.
- **Kiểm tra qua API:** Nếu bạn triển khai mô hình bằng cách sử dụng một API, bạn có thể sử dụng các công cụ như cURL, Postman hoặc thư viện requests của Python để gửi các yêu cầu HTTP đến các điểm cuối của API. Tạo yêu cầu với dữ liệu đầu vào cần thiết và gửi chúng đến API đã triển khai.

Ví dụ: Nếu bạn sử dụng Flask để tạo API, bạn có thể gửi một yêu cầu POST với dữ liệu kiểm tra là nội dung yêu cầu đến điểm cuối của API. Nhận phản hồi và phân tích các đầu ra dự đoán.

- **Đánh giá dự đoán:** So sánh các đầu ra dự đoán từ mô hình đã triển khai với nhãn thực tế hoặc kết quả dự đoán mong đợi cho các mẫu kiểm tra. Tính toán các chỉ số đánh giá liên quan (ví dụ: độ chính xác, độ chính xác, độ nhạy) để đánh giá hiệu suất của mô hình.
- **Phân tích lỗi:** Xác định các mẫu được phân loại sai hoặc các trường hợp mà mô hình không hoạt động tốt. Xem xét các mẫu được phân loại sai hoặc các trường hợp với độ tin cậy thấp để có cái nhìn về các vấn đề tiềm ẩn hoặc các khía cạnh cần cải thiện.
- **Kiểm tra hiệu suất:** Đo thời gian phản hồi và độ trễ của mô hình đã triển khai bằng cách gửi một số lượng lớn yêu cầu cùng một lúc hoặc liên tiếp. Điều này giúp đánh giá khả năng mở rộng và tính nhanh của mô hình dưới các tải khác nhau.
- **Ghi nhật ký và giám sát:** Triển khai cơ chế ghi nhật ký và giám sát để ghi lại thông tin quan trọng trong quá trình kiểm tra, chẳng hạn như nhật ký yêu



cầu, nhật ký lỗi và các chỉ số hiệu suất. Sử dụng các dịch vụ AWS CloudWatch hoặc các công cụ giám sát khác để theo dõi sức khỏe và hiệu suất của mô hình đã triển khai.

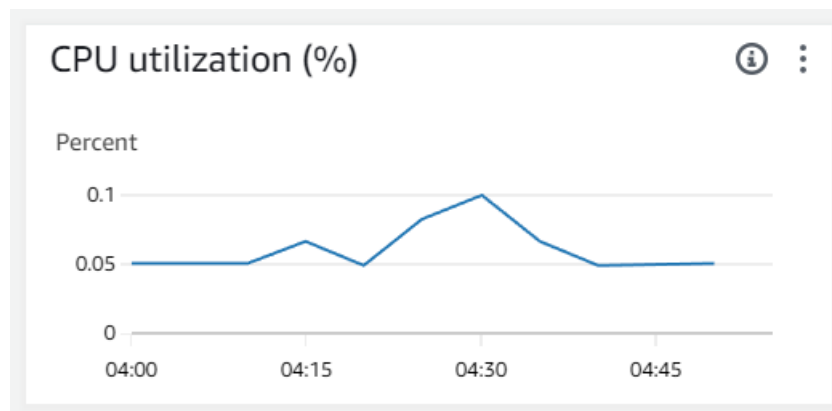
- **Phản hồi và lặp lại:** Dựa trên kết quả kiểm tra, điều chỉnh và lặp lại mô hình học máy nếu cần thiết. Kết hợp phản hồi từ giai đoạn kiểm tra vào quy trình phát triển của bạn để nâng cao khả năng của mô hình và giải quyết các vấn đề đã xác định.

Hãy nhớ ghi lại kết quả kiểm tra, bao gồm bất kỳ hạn chế hoặc khía cạnh cần cải thiện nào. Thực hiện kiểm tra định kỳ để đảm bảo tính chức năng và hiệu suất liên tục của mô hình học máy đã triển khai trên EC2 Instance trên AWS.

## 5.2 SỐ LIỆU HIỆU SUẤT

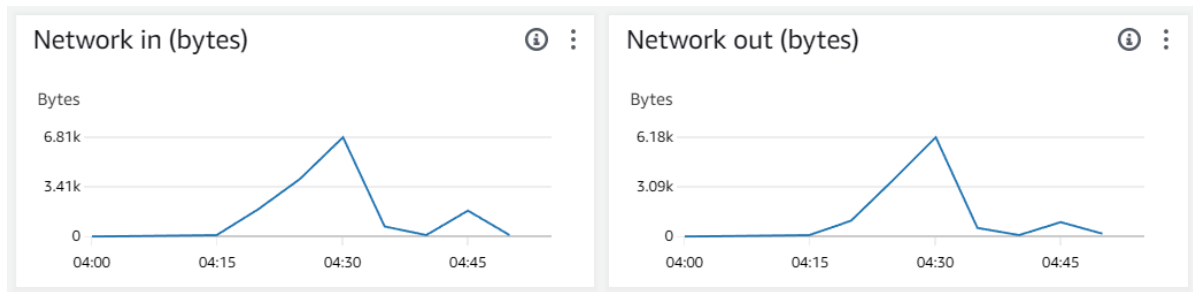
### ○ EC2 Instances:

CPU Utilization là một chỉ số quan trọng để đánh giá hiệu suất của các EC2 instances trên AWS. Nó cho biết tỉ lệ phần trăm của khả năng xử lý CPU đang được sử dụng. Khi CPU Utilization cao, điều này có thể chỉ ra rằng EC2 instance đang hoạt động gần giới hạn khả năng xử lý CPU của nó.



Network In và Network Out là hai chỉ số quan trọng để đo lường lưu lượng mạng đến và đi từ các EC2 instances trên AWS. Chúng giúp đánh giá hiệu suất và tài nguyên mạng của các instances.

Quan trọng nhất là giám sát và phân tích các chỉ số Network In và Network Out để đảm bảo rằng hệ thống mạng không bị bottleneck và có khả năng xử lý lưu lượng mạng đủ cho ứng dụng.

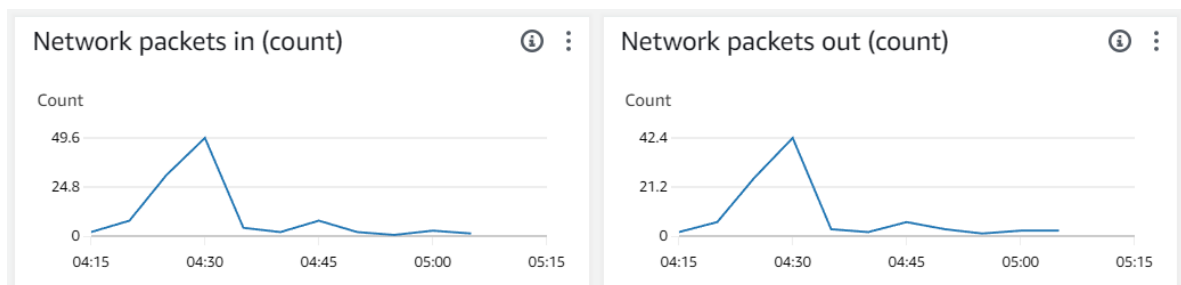


Network packets In/Out là thuật ngữ được sử dụng để đề cập đến số lượng gói tin mạng (network packets) được nhận và gửi đi từ các instances EC2.

Khi một instance EC2 nhận một gói tin từ mạng, nó được tính là một network packet In. Đây là gói tin được gửi đến instance từ bên ngoài, chẳng hạn như yêu cầu HTTP từ một người dùng hoặc phản hồi từ một dịch vụ khác trên mạng.

Ngược lại, khi một instance EC2 gửi một gói tin đi ra mạng, nó được tính là một network packet Out. Đây là gói tin được gửi từ instance đến một đích bên ngoài, chẳng hạn như phản hồi HTTP từ instance đến người dùng hoặc yêu cầu truy vấn cơ sở dữ liệu từ instance đến một dịch vụ cơ sở dữ liệu trên mạng.

Giám sát số lượng network packets In/Out trong EC2 có thể giúp đánh giá lưu lượng mạng đang diễn ra trên instances và xác định tải trọng mạng hiện tại của hệ thống. Bằng cách theo dõi và phân tích các chỉ số này, có thể đảm bảo rằng mạng không bị quá tải và ứng dụng có đủ khả năng xử lý lưu lượng mạng.



Trong dịch vụ EC2 của AWS, CPU credit usage và CPU credit balance là các thuật ngữ được sử dụng để đo lường và quản lý việc sử dụng tài nguyên CPU trên các instance của EC2, đặc biệt đối với các loại instance được gọi là "burstable performance instances".

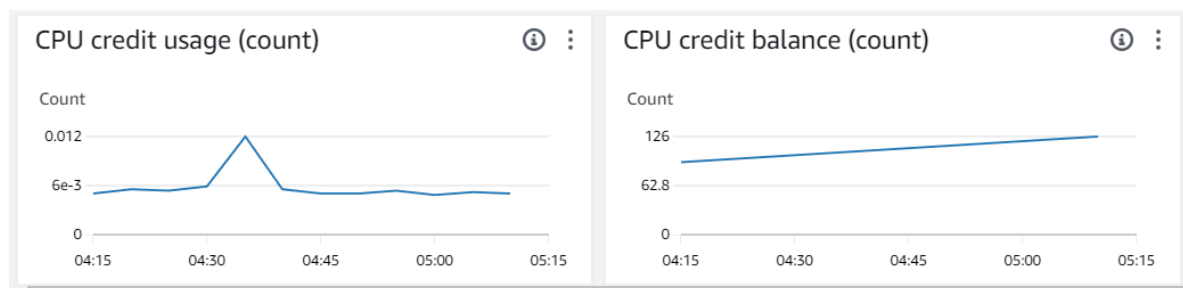
Các loại instance burstable performance, như t2, t3 và t4g, có khả năng sử dụng CPU vượt quá năng lực cơ bản được cấp cho chúng trong một khoảng thời gian ngắn, nhờ vào việc tích lũy CPU credits. CPU credits là một đơn vị đo lường tài nguyên CPU được tích lũy theo thời gian và sử dụng để quy định khả năng CPU

burst của các instance.

Dưới đây là hai thuật ngữ quan trọng liên quan đến CPU credit trong EC2:

- CPU credit balance: Đây là số lượng CPU credits hiện có trong tài khoản của bạn. Khi instance không sử dụng hết CPU credits được cấp, nó sẽ tích lũy thành một số lượng CPU credit balance. Số lượng CPU credit balance này có thể được sử dụng trong tương lai để cho phép instance thực hiện các hoạt động burst CPU khi cần thiết.
- CPU credit usage: Đây là số lượng CPU credits đã sử dụng bởi instance trong một khoảng thời gian nhất định. Khi instance hoạt động trong giới hạn của năng lực cơ bản của nó, CPU credit sẽ không bị sử dụng. Tuy nhiên, khi instance vượt quá năng lực cơ bản, CPU credits sẽ được sử dụng để duy trì hiệu suất burst trong một thời gian ngắn. CPU credit usage sẽ tăng khi instance đang sử dụng năng lực CPU vượt quá giới hạn của năng lực cơ bản.

Quản lý CPU credit balance và CPU credit usage là quan trọng để đảm bảo rằng các instance burstable performance của bạn có đủ CPU credits để sử dụng khi cần thiết. Nếu CPU credit balance cạn kiệt, instance có thể gặp giới hạn về hiệu suất và không thể duy trì tốc độ burst CPU. Do đó, nếu bạn sử dụng các loại instance burstable performance, hãy theo dõi và quản lý CPU credit balance và CPU credit usage để đảm bảo rằng ứng dụng của bạn hoạt động ổn định và không bị giới hạn bởi CPU credits.



## CHƯƠNG 6

# PHÂN TÍCH CÁC DỊCH VỤ VÀ CÔNG NGHỆ Đám Mây

### 6.1 THÔNG TIN CHI TIẾT VỀ DỊCH VỤ

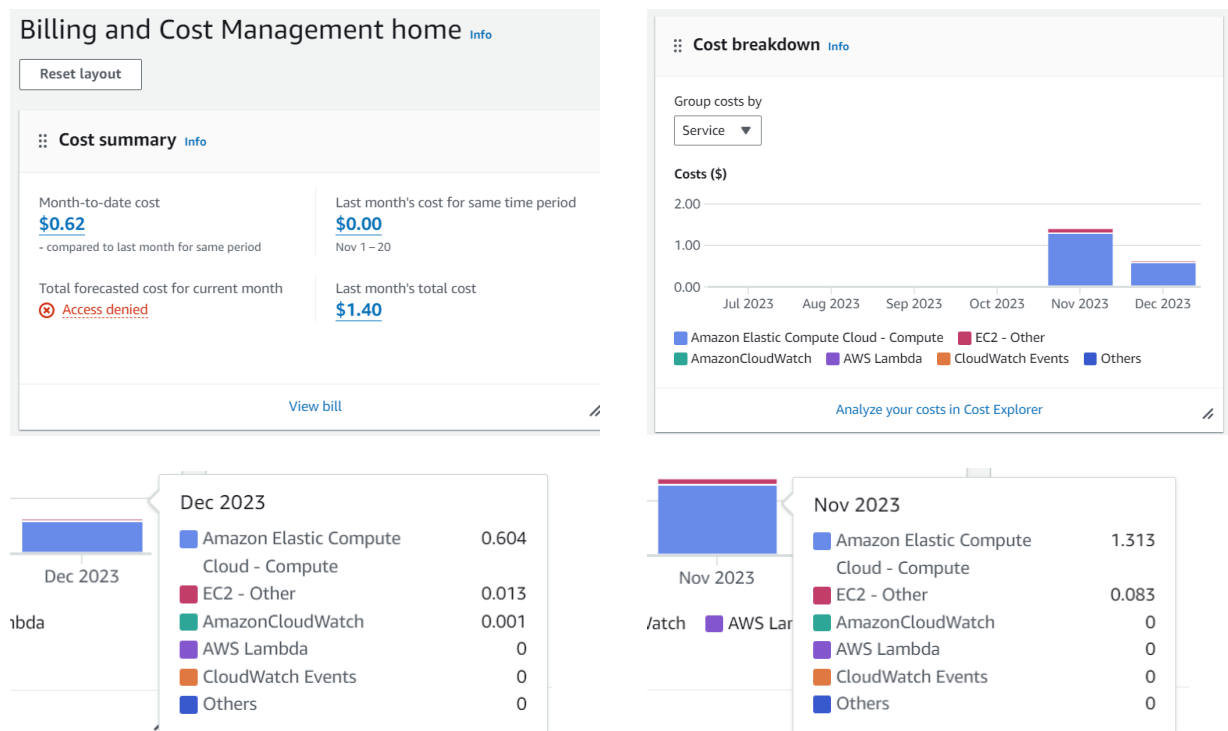
Dịch vụ EC2 (Elastic Compute Cloud) là một trong những dịch vụ quan trọng và phổ biến nhất trong Amazon Web Services (AWS). Nó cung cấp một môi trường linh hoạt để triển khai và quản lý các máy ảo trên đám mây của AWS. Dưới đây là một số thông tin chi tiết về dịch vụ EC2:

1. **Instances:** EC2 cho phép bạn tạo và quản lý các instances (máy ảo) trên đám mây của AWS. Bạn có thể lựa chọn từ nhiều loại instance với các thông số về khả năng xử lý, bộ nhớ, lưu trữ và mạng khác nhau để phù hợp với nhu cầu của ứng dụng của bạn.
2. **Elasticity:** EC2 cho phép mở rộng hoặc thu hẹp số lượng instances một cách linh hoạt tùy thuộc vào nhu cầu của bạn. Bạn có thể sử dụng dịch vụ Auto Scaling để tự động thay đổi số lượng instances dựa trên các ngưỡng như tải CPU, lưu lượng mạng hoặc các yếu tố khác.
3. **Lưu trữ:** EC2 cung cấp nhiều tùy chọn lưu trữ cho instances của bạn. Bạn có thể sử dụng các tùy chọn như Amazon EBS (Elastic Block Store) để lưu trữ dữ liệu liên tục, Amazon S3 (Simple Storage Service) để lưu trữ đối tượng, hoặc EFS (Elastic File System) để chia sẻ dữ liệu giữa nhiều instances.
4. **Bảo mật:** EC2 cung cấp các tính năng bảo mật để bảo vệ instances và dữ liệu của bạn. Bạn có thể sử dụng các em bảo mật (security groups) để kiểm soát quyền truy cập vào instances, sử dụng khóa SSH để truy cập từ xa vào instances và cung cấp các biện pháp bảo mật khác như IAM (Identity and Access Management), VPC (Virtual Private Cloud) để cô lập mạng.
5. **Quản lý:** EC2 cung cấp các công cụ quản lý mạnh mẽ để giúp bạn quản lý và theo dõi instances của mình. Bạn có thể sử dụng AWS Management Console, AWS Command Line Interface (CLI) hoặc API để tạo, cấu hình và giám sát instances. EC2 cũng tích hợp với nhiều dịch vụ khác của AWS như CloudWatch, CloudTrail và Systems Manager để giúp bạn quản lý và giám sát tài nguyên EC2 một cách hiệu quả.

- Định vị địa lý: EC2 cho phép bạn triển khai instances ở nhiều vị trí địa lý khác nhau trên toàn cầu. Bằng cách chọn các khu vực (regions) và các vùng khả dụng (availability zones), bạn có thể đảm bảo tính sẵn sàng và độ tin cậy cao cho ứng dụng của mình thông qua việc phân tán và sao lưu dữ liệu.

Đây là một số thông tin cơ bản về dịch vụ EC2 trên AWS. EC2 cung cấp một môi trường linh hoạt và mạnh mẽ để triển khai và quản lý các máy ảo trên đám mây, phục vụ nhu cầu của các ứng dụng từ nhỏ đến lớn.

## 6.2 PHÂN TÍCH CHI PHÍ



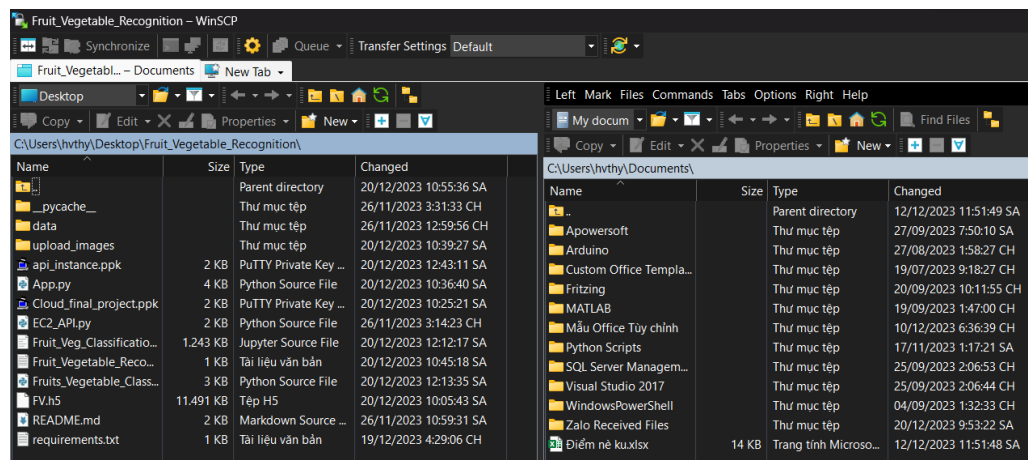
- Chi phí tính đến thời điểm hiện tại trong tháng là 0.62\$  
Trong đó bao gồm 0.604\$ cho Amazon Elastic Compute Cloud và EC2 0.013\$  
Và 0.001\$ cho AmazonCloudWatch
- Tổng chi phí của tháng trước là 1.4\$  
Trong đó bao gồm 1.313\$ cho Amazon Elastic Compute Cloud và EC2 0.083\$

# CHƯƠNG 7

## KẾT LUẬN

### 7.1 KẾT QUẢ ĐẠT ĐƯỢC

- Truyền thành công tệp tin từ máy tính nội bộ sang một instance EC2 sử dụng WinSCP



- Kết nối thành công với một instance EC2 bằng PuTTY

```
ubuntu@ip-172-31-62-68: ~/Fruit_Vegetable_Recognition
login as: ubuntu
Authenticating with public key "Cloud_final_project"
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1048-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Wed Dec 20 05:38:20 UTC 2023

System load:  0.0               Processes:    108
Usage of /:   57.7% of 7.57GB   Users logged in: 0
Memory usage: 5%               IPv4 address for eth0: 172.31.62.68
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

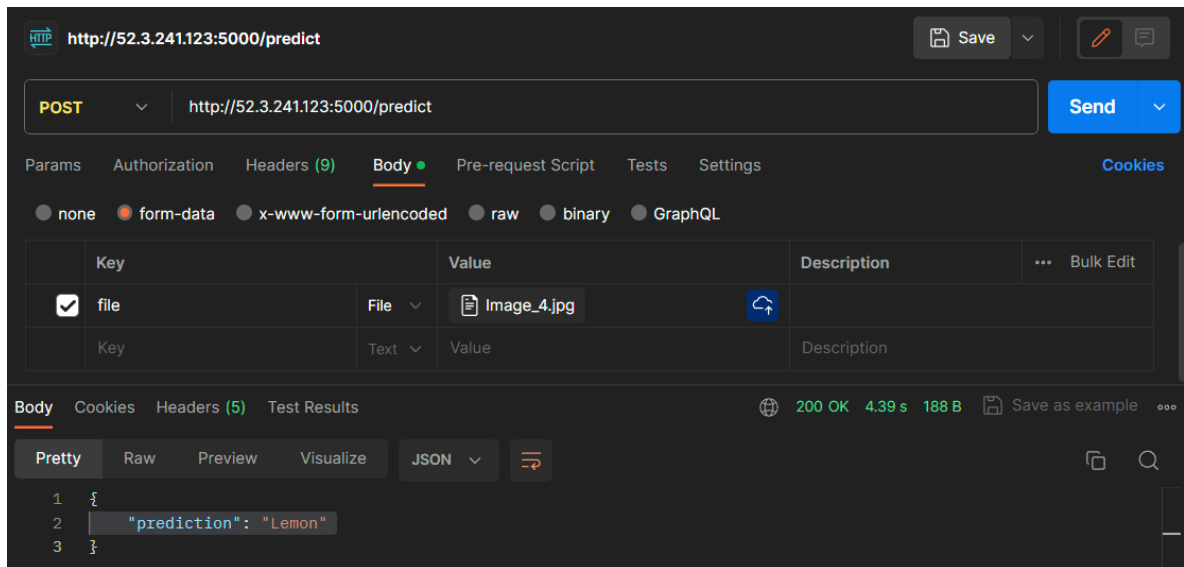
52 updates can be applied immediately.
40 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

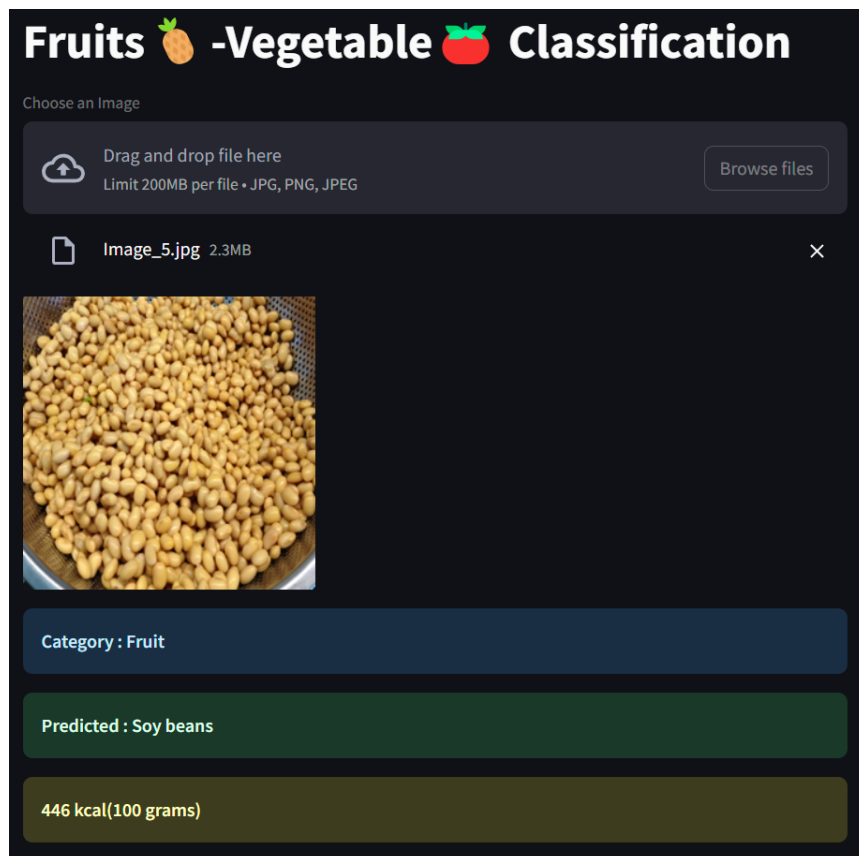
New release '22.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Wed Dec 20 03:39:59 2023 from 1.53.41.210
```

- Triển khai mô hình trên EC2 và sử dụng Postman để nhập dữ liệu, thực hiện dự đoán



- Triển khai mô hình trên EC2 và sử dụng Streamlit để hiển thị giao diện người dùng



## 7.2 PHÁT TRIỂN TRONG TƯƠNG LAI

Trong tương lai, quy trình phát triển tự động sẽ sử dụng AWS CodePipeline và AWS CodeCommit để xây dựng hình ảnh Docker trên AWS CodeBuild sẽ tiếp tục phát triển và được tối ưu hóa để đáp ứng các yêu cầu về tính khả dụng và hiệu suất.

Quy trình làm việc sẽ sử dụng AWS CodeCommit làm kho lưu trữ mã nguồn, trong đó nhà phát triển ứng dụng sẽ đẩy mã mới vào. Sau đó, AWS CodeBuild sẽ tự động xây dựng lại mã từ kho lưu trữ và cập nhật dịch vụ.

Để đảm bảo tính khả dụng cao, quy trình cũng sẽ sử dụng AWS Application Load Balancer để phân phối tải đến các phiên bản ứng dụng. Load balancer này sẽ tự động điều chỉnh quy mô của phiên bản ứng dụng dựa trên yêu cầu và tải hiện tại, đảm bảo rằng ứng dụng luôn sẵn sàng và phản hồi nhanh chóng đối với người dùng.

Để theo dõi và quản lý hiệu suất, AWS CloudWatch sẽ cung cấp tính năng ghi nhật ký và cảnh báo. CloudWatch sẽ giúp theo dõi các chỉ số và sự cố của ứng dụng, cung cấp thông tin quan trọng để xử lý vấn đề và đảm bảo rằng mục tiêu hiệu suất được đáp ứng.

Cuối cùng, để đảm bảo rằng phiên bản mới nhất của ứng dụng được triển khai, quy trình sẽ đảm bảo rằng mẫu Docker mới nhất được chọn từ kho lưu trữ S3, nơi chứa các phiên bản hình ảnh Docker đã được xây dựng.

Tổng quan, quy trình phát triển tự động này kết hợp các dịch vụ và công cụ của AWS để tạo ra một quy trình phát triển liên tục và đáng tin cậy, đáp ứng các yêu cầu về tính khả dụng và hiệu suất của ứng dụng.



## TÀI LIỆU THAM KHẢO

[1]. Cloud report template,

[https://docs.google.com/document/d/1x-\\_OIKCU5ihQ2i0JiQi3xFhtRVW-jDdAvb-BXRLFUAo/edit?usp=sharing](https://docs.google.com/document/d/1x-_OIKCU5ihQ2i0JiQi3xFhtRVW-jDdAvb-BXRLFUAo/edit?usp=sharing).

[2]. AWS tutorials,

<https://aws.amazon.com/tutorials/train-deep-learning-model-aws-ec2-containers/>

[3]. Medium Blog,

<https://medium.com/shapeai/deploying-flask-application-with-ml-models-on-aws-ec2-instance-3b9a1cec5e13>

[4]. Youtube Video Example,

[https://youtu.be/oHHxXZLppZs?si=SGT-xzx5lcvM\\_wm9](https://youtu.be/oHHxXZLppZs?si=SGT-xzx5lcvM_wm9)

## PHỤ LỤC

[1]. Source code,

<https://github.com/xiisii/Cloud-AWS-Project>

[2]. Architecture,

[https://drive.google.com/file/d/1jws5MaJ2zyJzlee7-wZrKxN9GsE8gUMV/view?usp=drive\\_link](https://drive.google.com/file/d/1jws5MaJ2zyJzlee7-wZrKxN9GsE8gUMV/view?usp=drive_link)

[3]. Datasets,

<https://www.kaggle.com/datasets/kritikseth/fruit-and-vegetable-image-recognition>

[4]. Video Demo,

[https://drive.google.com/file/d/1KrX-rGNciZUBVQf3xgupgO-btS1AB\\_O/view?usp=drive\\_link](https://drive.google.com/file/d/1KrX-rGNciZUBVQf3xgupgO-btS1AB_O/view?usp=drive_link)

[5]. PAQ-Answer,

[https://docs.google.com/document/d/1rKBGVlShglpoBhMvJ-ztVdKuQJo\\_cEYH/edit?usp=drive\\_link&ouid=115542434220792225785&rtpof=true&sd=true](https://docs.google.com/document/d/1rKBGVlShglpoBhMvJ-ztVdKuQJo_cEYH/edit?usp=drive_link&ouid=115542434220792225785&rtpof=true&sd=true)