Trường Đại học Công nghệ Thông tin
ĐHQG-HCM

Tháng 5, 2025

Báo cáo cuối kỳ

MLOPS: TỰ ĐỘNG HÓA TRIỂN KHAI PHÂN TÍCH CẢM XÚC VỚI SAGEMAKER, GITHUB ACTIONS VÀ CLOUDFORMATION

Nhóm thực hiện: Nhóm 21

Lê Bình Nguyên - 22520969
Đặng Hữu Phát - 22521065
Châu Thế Vĩ - 22521653

Mục lục

1. GIỚI THIỆU

1. Giới thiệu

DevOps

DevOps: Phương pháp luận kết hợp ***Phát triển (Development)*** và ***Vận hành (Operation)***, tăng cộng tác, rút ngắn vòng đời phát triển, nâng cao chất lượng, giảm thời gian ra thị trường.

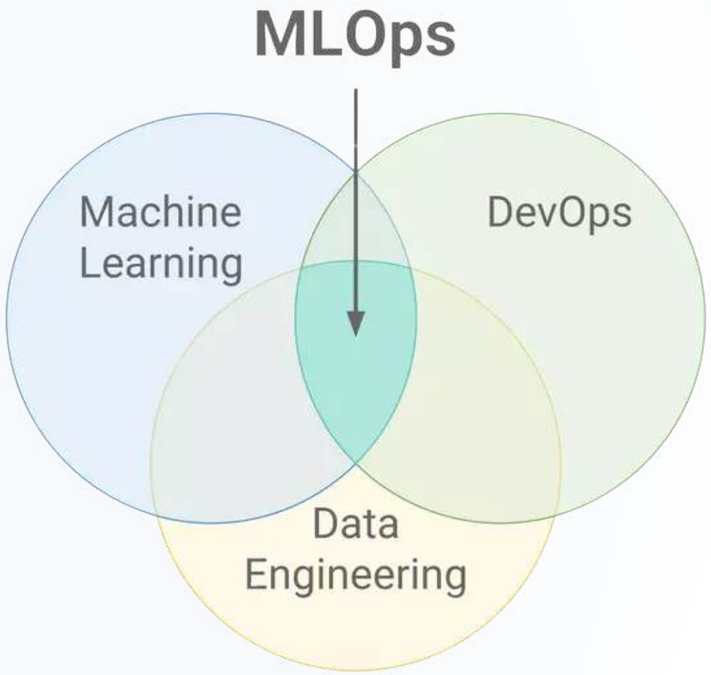
- **Tự động hóa:** xây dựng, kiểm thử, triển khai, giám sát.
- **Lợi ích:** giảm lỗi, cải thiện hiệu suất, linh hoạt trước thay đổi.
- **Trong đồ án này:** DevOps là *"xương sống"*, xây dựng pipeline hiệu quả, hỗ trợ tích hợp, kiểm thử, triển khai mô hình học máy liên tục.

1. Giới thiệu

MLOps

MLOps(Machine Learning Operations):
Nhánh mở rộng của DevOps, chuyên quản lý hệ thống học máy.

- **Kết hợp:** DevOps, khoa học dữ liệu, AI.
- **Mục tiêu:** Tự động hóa, kiểm soát vòng đời mô hình (thu thập dữ liệu, huấn luyện, triển khai, cập nhật).
- **Lợi ích:** Tăng hiệu quả, đảm bảo chất lượng, dễ bảo trì.
- **Trong dự án này:** Hỗ trợ triển khai, giám sát, cập nhật mô hình phân tích cảm xúc liên tục.



1. Giới thiệu

MLOps

Trong một dự án **phân tích cảm xúc**, MLOps đóng vai trò quan trọng bằng cách:

- **Đảm bảo triển khai nhất quán:** Giúp mô hình hoạt động ổn định khi đưa vào thực tế.
- **Kiểm soát chất lượng và hiệu suất:** Theo dõi để đảm bảo mô hình luôn chính xác và hiệu quả.
- **Hỗ trợ cập nhật liên tục:** Cho phép điều chỉnh mô hình dễ dàng khi dữ liệu hoặc yêu cầu thay đổi.

1. Giới thiệu

Về đề tài

Phân tích cảm xúc là quá trình sử dụng học máy để xác định cảm xúc (tích cực, tiêu cực) từ văn bản. Trong đề tài này, dự án tập trung xây dựng một hệ thống triển khai mô hình học máy hoàn chỉnh, phân tích cảm xúc từ nhận xét phim trong bộ dữ liệu IMDB.

1. Giới thiệu

Về đề tài

INPUT

This film is so good <3

Một câu đánh giá phim

Public Model

OUTPUT

tích cực/tiêu cực

Phân loại

1. Giới thiệu

Mục tiêu

Mục tiêu dự án: Xây dựng hệ thống triển khai mô hình học máy phân tích cảm xúc từ dữ liệu văn bản IMDB.

Thành phần:

- Pipeline tự động hóa: Huấn luyện, đóng gói, triển khai mô hình.
- API công khai: Dự đoán cảm xúc theo thời gian thực.
- Web frontend: Giao diện thân thiện, tích hợp API.
- Giám sát & mở rộng: Linh hoạt, dễ bảo trì.

Thiết kế: Dựa trên DevOps và MLOps, đảm bảo bảo mật, khả năng mở rộng, dễ bảo trì, và giám sát chặt chẽ.

2. CÔNG NGHỆ SỬ DỤNG

a. Nền tảng Đám mây & Cơ sở hạ tầng

AWS SageMaker

- Nền tảng học máy end-to-end của AWS
- Huấn luyện, phát triển và triển khai mô hình
- Use case: Huấn luyện và phát triển mô hình phân tích cảm xúc

Amazon S3

- Amazon Simple Storage Service
- Lưu trữ dữ liệu huấn luyện, model artifacts (tệp mô hình) và mã nguồn
- Use cases:
  - Lưu trữ dữ liệu huấn luyện
  - Lưu trữ model artifacts
  - Lưu trữ mẫu cấu hình hạ tầng AWS
  - Mã nguồn cho các hàm AWS Lambda

a. Nền tảng Đám mây & Cơ sở hạ tầng

Amazon ECR

- Elastic Container Registry
- Store và manage Docker images
  - Use cases:
  - SageMaker training containers
  - SageMaker inference containers
  - Custom model containers

AWS Lambda

- Nền tảng serverless, chạy mã không cần quản lý máy chủ.
- Xử lý yêu cầu API.
- Định dạng đầu vào cho SageMaker Endpoint.
- Trả kết quả suy luận về client.

Amazon API Gateway

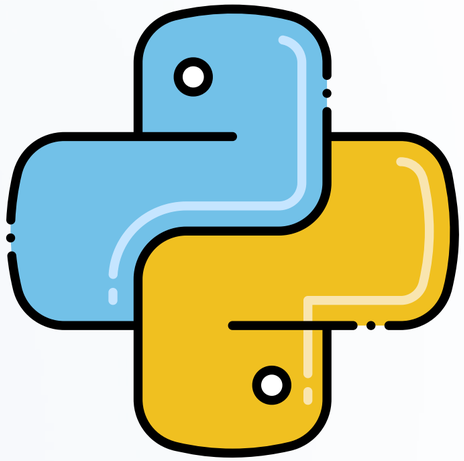
- Dịch vụ quản lý API RESTful/WebSocket.
- Kết nối frontend với Lambda.
- Bảo mật, kiểm soát quyền truy cập.
- Theo dõi lưu lượng, phân tích log.

b. Khung học máy và thư viện

Python

Ngôn ngữ lập trình chính được sử dụng xuyên suốt toàn bộ dự án, nhờ vào:

- Cú pháp đơn giản, dễ đọc, dễ duy trì.
- Thư viện học máy và xử lý dữ liệu phong phú.
- Hỗ trợ tốt cho container hóa và tích hợp với các dịch vụ AWS.



b. Khung học máy và thư viện

PyTorch

PyTorch là một framework học sâu mã nguồn mở, nổi bật với:

- Cơ chế mô hình hóa động giúp dễ dàng debug.
- Tối ưu tốt trên GPU.
- Cộng đồng mạnh và tài liệu đầy đủ.
- Trong dự án, PyTorch được sử dụng để xây dựng **mô hình LSTM** phân tích cảm xúc.

b. Khung học máy và thư viện

scikit-learn

skicit-learn là thư viện cổ điển cho các thuật toán học máy, tiền xử lý và đánh giá mô hình.

Ứng dụng trong dự án:

- Chuẩn hóa dữ liệu đầu vào.
- Đánh giá mô hình với các chỉ số như Accuracy, F1-score.
- Hỗ trợ kiểm thử và so sánh nhiều pipeline

c. Quản lý hạ tầng bằng mã (IaC)

AWS CloudFormation

CloudFormation cho phép định nghĩa hạ tầng AWS bằng tệp YAML hoặc JSON để triển khai tự động.

Lợi ích:

- Giảm lỗi cấu hình thủ công.
- Dễ dàng mở rộng và tái sử dụng.
- Tự động hóa toàn bộ môi trường với file cấu hình duy nhất.

d. Tự động hóa CI/CD

GitHub Actions

GitHub Actions là nền tảng tự động hóa quy trình ngay trong kho mã nguồn GitHub.
Trong dự án, GitHub Actions giúp:

- Kiểm thử mã tự động khi có pull request.
- Build và đẩy Docker image lên ECR.
- Triển khai mô hình và API.
- Deploy giao diện web lên S3

e. Container hoá

Docker

Docker là công cụ container hóa ứng dụng phổ biến, được sử dụng để:

- Đóng gói môi trường huấn luyện và inference.
- Tái sử dụng mô hình dễ dàng qua các bước.
- Đảm bảo môi trường thống nhất giữa dev và prod

e. Giám sát và ghi log

AWS CloudWatch

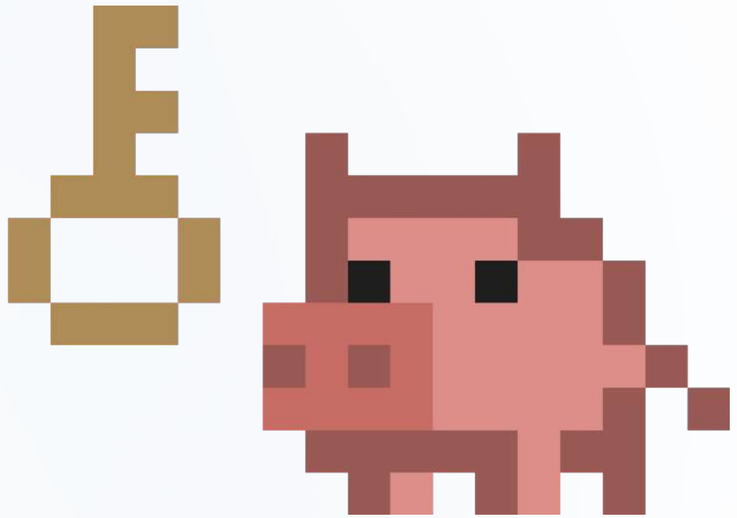
CloudWatch cho phép thu thập log, giám sát hiệu năng và tạo cảnh báo. Trong dự án, nó dùng để:

- Theo dõi job huấn luyện trên SageMaker.
- Ghi log từ Lambda và các lỗi hệ thống.
- Hỗ trợ cảnh báo sớm và phân tích sự cố.

f. Phân tích bảo mật mã nguồn

TruffleHog

- **TruffleHog** là công cụ tìm kiếm thông tin nhạy cảm như API key, secret token, mật khẩu,... nhằm tránh rò rỉ dữ liệu bảo mật
- **Cài đặt:** pip install trufflehog3
- **TruffleHog** là lựa chọn lý tưởng để bảo vệ dữ liệu trong dự án, dễ tích hợp vào CI/CD. Xem thêm tại GitHub.



f. Phân tích bảo mật mã nguồn

Bandit

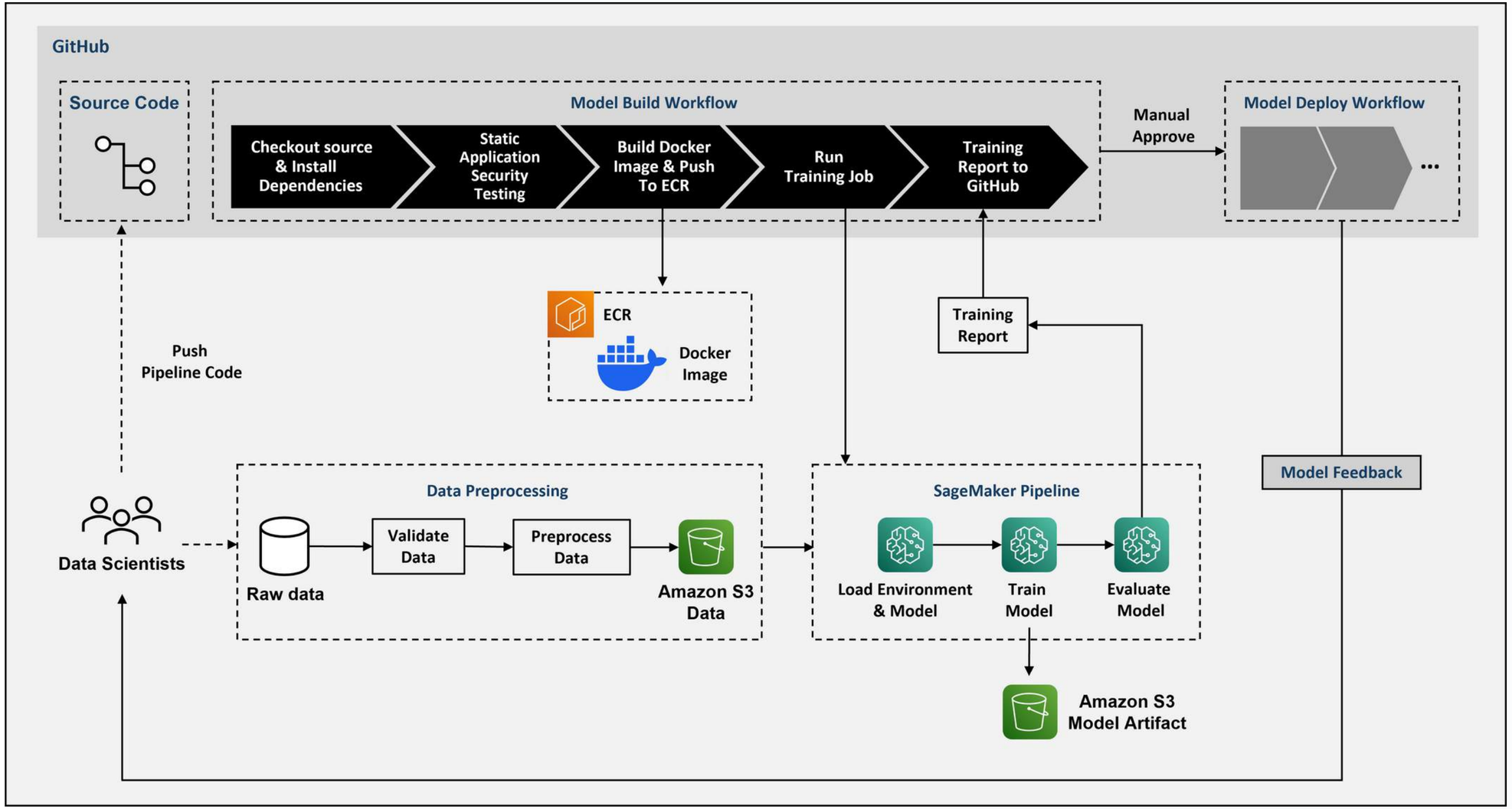
Bandit là công cụ phân tích tĩnh mã nguồn Python, phát hiện các vấn đề bảo mật phổ biến như:

- Sử dụng hàm eval không an toàn.
- Lỗi thao tác file hoặc xử lý input từ người dùng.
- Lạm dụng thư viện hoặc cấu hình sai.

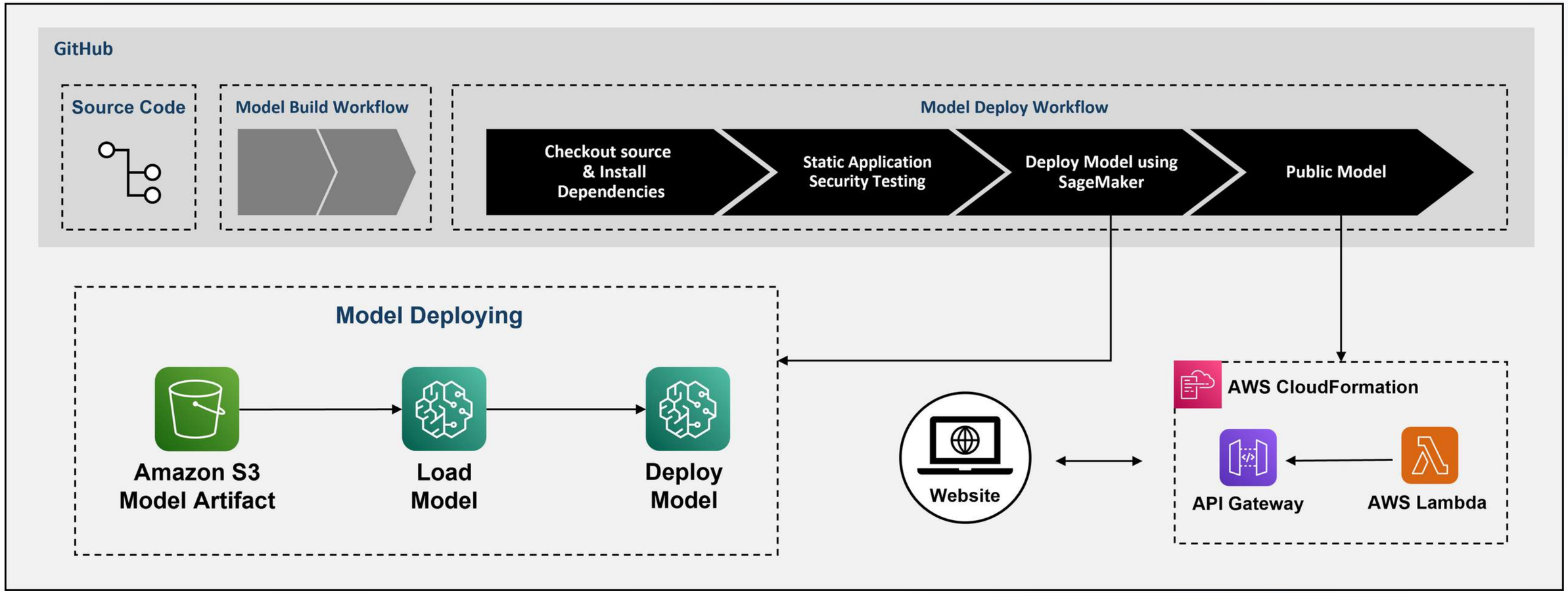
Xem thêm tại GitHub.

3. TRIỂN KHAI HỆ THỐNG

a. Quá trình huấn luyện, đánh giá mô hình



b. Quá trình triển khai, public mô hình



4. DEMO

GitHub

https://github.com/lbngyn/sagemaker-deployment

Demo

https://youtu.be/8Enjsm-Efes?si=FpSIS6Bor-x7f9Om

5. TỔNG KẾT

a. Kết quả đạt được

- Áp dụng thành công DevOps trong triển khai mô hình học máy trên AWS.
- Sử dụng hiệu quả các công cụ:
  - GitHub Actions (CI/CD)
  - Amazon SageMaker (triển khai mô hình)
  - AWS CloudFormation (quản lý hạ tầng)
  - Amazon S3 (lưu trữ mô hình, website)
  - Bandit, TruffleHog (kiểm tra bảo mật)
- Triển khai mô hình phân tích cảm xúc qua API Gateway và Lambda, tích hợp giao diện web.
- Đảm bảo quy trình mở rộng và tái sử dụng.

b. Hạn chế và cải thiện

- Bảo mật chưa toàn diện.
- Chưa triển khai lên nhiều môi trường
- Chưa tích hợp DataOps để tự động hóa khi dữ liệu thay đổi.
- Thiếu giám sát hệ thống (Prometheus + Grafana).
- Giao diện web đơn giản, cần phát triển dashboard tương tác.

c. Định hướng tương lai

- Tích hợp CI/CD với DataOps.
- Tăng cường bảo mật ứng dụng và API.
- Tích hợp giám sát và cảnh báo tự động.
- Mở rộng frontend (React/Vue) cho tương tác người dùng.

Tóm tắt

- Đồ án củng cố kiến thức DevOps trong triển khai học máy.
- Tạo nền tảng cho các dự án AI quy mô lớn và chuyên nghiệp.

CẢM ƠN

THẦY VÀ CÁC BẠN ĐÃ XEM!