I190726 Malaika Waheed
I192187 Asfandyar Sabri

MLOps Project Report

Project Name: Cryptographically Secured Healthcare Consortia for Federated Learning setup

Introduction

This project is based upon the concept of Federated Learning. Federated Learning is a technique that enables a large number of users to locally train a model on local datasets. The knowledge (trained weights) of these models is then shared on a network and it is aggregated into a bigger model. All of this is done without sharing any data. Though FL ensures that data remains anonymous to the silos part of the system, there is still a risk of various privacy attacks during the training process. The proposed solution in this project is an encryption algorithm as a measure for privacy preservation in a FL model. A **horizontal cross silo Federated Learning model** is developed that provides prediction for **chest diseases** in a healthcare setup. CNN model is used to train data on individual nodes.

This report gives an overview of the steps taken to automate the processes involved in this project. Various MLOps techniques have been applied to integrate and maintain machine learning models efficiently.

Data Version Control (DVC)

DVC has been used to version control project files. To implement federated learning weights are extracted from all the models from their respective clients. The weights are stored in .pth files. They are usually very large files and are updated in each iteration of federated learning. In this project three clients are taken for demonstration and therefore there are three .pth files. These are stores on google drive using dvc. The steps to add dvc files are as follows

```
dvc init
dvc remote add mlops gdrive://1PkU39Mqsb_dB343iGaDefd4HSvm_AJmz
dvc add model1.pth (model2.pth, model3.pth)
dvc push -r mlops
```

To pull these file use the command

```
dvc pull
```

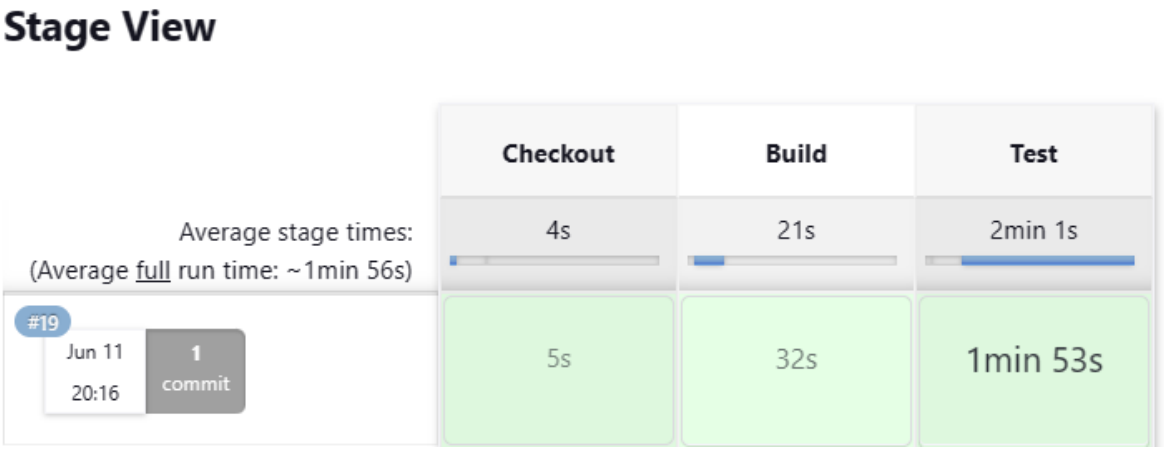
I190726 Malaika Waheed
I192187 Asfandyar Sabri

Jenkins CI/CD Pipeline

Jenkins has collaborated with Github using Poll SCM. Jenkins after a set interval checks the Github repository. If any changes are made to the Github repository, a build is triggered in Jenkins. The pipeline for Jenkins is as follows:

```
pipeline {
    agent any

    stages {
        stage('Checkout') {
            steps {
                git branch: 'Malaika', url: 'https://github.com/Malaika01/MLOps-Project.git'
            }
        }

        stage('Build') {
            steps {
                script {


                    // Execute the pip install command
                    bat 'pip install tqdm numpy torch medmnist torchvision dataclasses pylint'

                    // Run your Python script or commands here
                      bat "pylint --disable=C,R,W0104 cryptogram_final.py"

                }
            }
        }
        stage('Test') {
            steps {
                script {

                    // Execute the pip install command
                    bat 'python unit_tests.py'

                }
            }
        }
    }
}
```

I190726 Malaika Waheed
I192187 Asfandyar Sabri



Docker Image

Docker is used to containerize the files. An image of the cryptogram.py file which includes the code of CNN model training and encryption of the weights of CNN model is created. The image is used to create containers. The following are the commands for this

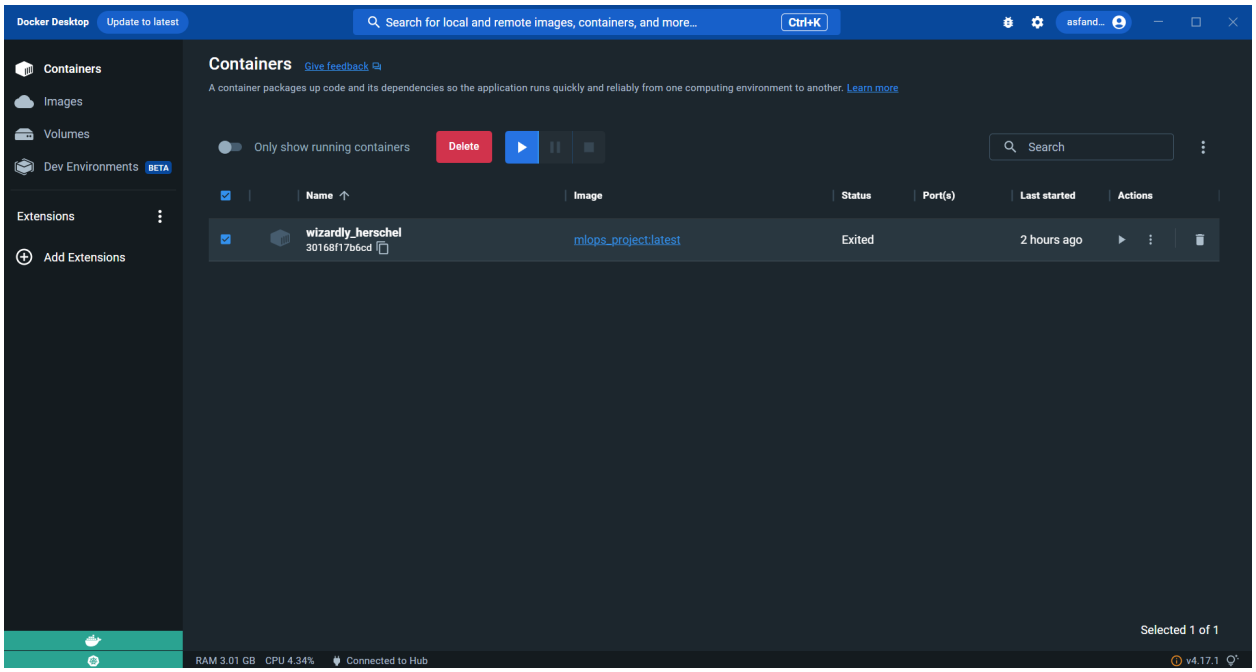
docker build -t mlops_project
docker run <id>

The dockerfile is as follows:

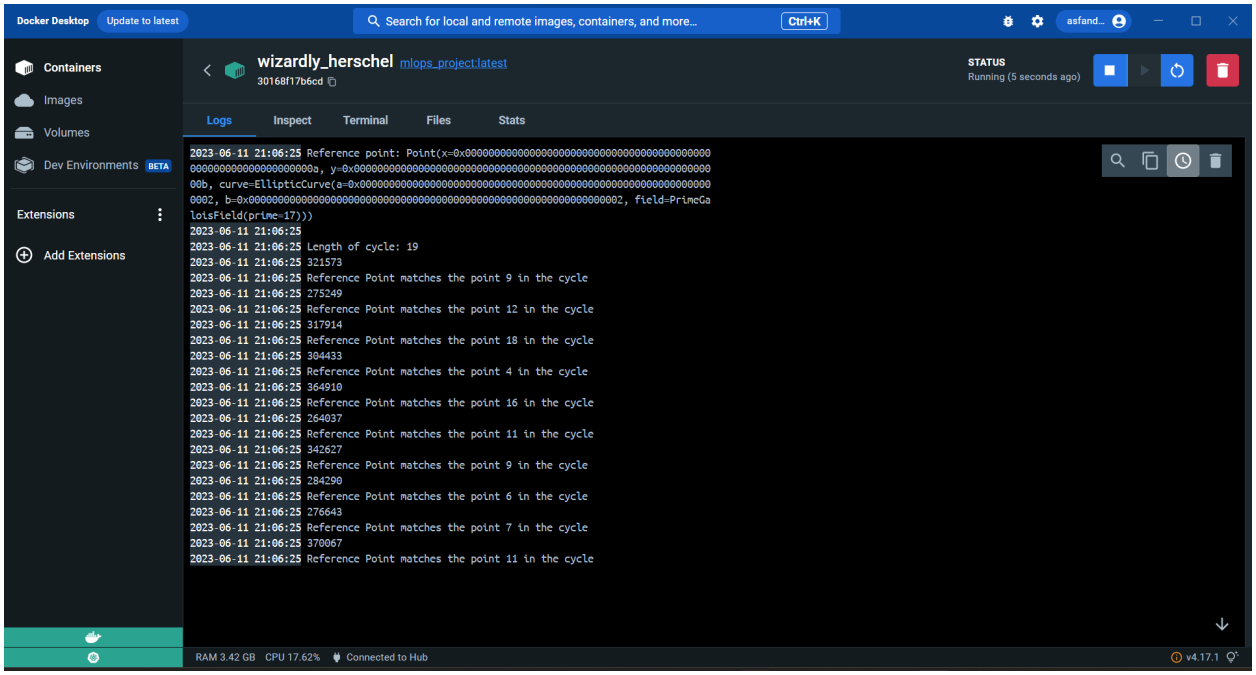
```
# Use a base image with Python and necessary dependencies
FROM python:3.9

# Set the working directory inside the container
WORKDIR /app

# Copy the entire project directory into the container at /app
COPY . /app

# Install any dependencies required by your project
RUN pip install -r requirements.txt

# Specify the command to run your Python script
CMD ["python", "cryptogram_final.py"]
```

I190726 Malaika Waheed
I192187 Asfandyar Sabri

Container:



Container running:



I190726 Malaika Waheed
I192187 Asfandyar Sabri

Dockerfile:

```
# Use a base image with Python and necessary dependencies
FROM python:3.9

# Set the working directory inside the container
WORKDIR /app

# Copying the entire project directory into the container at /app
COPY . /app

# Installing any dependencies required by your project
RUN pip install -r requirements.txt

# Specifying the command to run Python script
CMD ["python", "cryptogram_final.py"]
```

Mlflow

Mlflow is used to track the project. The parameters and metrics of individual and aggregated models are logged and compared.

Logs for model 1

I190726 Malaika Waheed
I192187 Asfandyar Sabri

```
==> Evaluating ...
train  auc: 0.584  acc:0.949
test  auc: 0.580  acc:0.947
torch.Size([16, 1, 3, 3])
torch.Size([16])
torch.Size([16])
torch.Size([16])
torch.Size([16, 16, 3, 3])
torch.Size([16])
torch.Size([16])
torch.Size([16])
torch.Size([64, 16, 3, 3])
torch.Size([64])
torch.Size([64])
torch.Size([64])
torch.Size([64, 64, 3, 3])
torch.Size([64])
torch.Size([64])
torch.Size([64])
torch.Size([64, 64, 3, 3])
torch.Size([64])
torch.Size([64])
torch.Size([64])
torch.Size([128, 1024])
torch.Size([128])
torch.Size([128, 128])
torch.Size([128])
torch.Size([14, 128])
torch.Size([14])
```