Threat Fabric MLOPS Challenge

The goal of this assignment is to assess how much you are expertise in developing pipeline to build and deploy a machine learning solution. Hence, we would like to know the detailed steps you follow during the implementation.

Description: In this assignment you are requested to build and deploy a Machine learning model for user recognition based on keystroke data and is consists of 2 separate parts.

Part 1 – Building Model

The objective of this part is building ML models for user recognition based on their keystroke data. keystroke dynamics is a behavioural biometric which utilizes the unique way a person types to verify the identity of an individual. Typing patterns are predominantly extracted from computer keyboards. the patterns used in keystroke dynamics are derived mainly from the two events that make up a keystroke: the Key-Press and Key-Release. The Key-Press event takes place at the initial depression of a key and the Key-Release occurs at the subsequent release of that key.

In this step, a dataset of keystroke information of users is given with following information:

- Train_keystroke.csv: in this dataset the keystroke data from 110 users are collected. All users are asked to type a 13-length constant string 8 times and the keystroke data (key-press time and key-release time for each key) are collected. The data set contains 880 rows and 27 columns. The first column indicates UserID, and the rest shows the press and release time for first to 13th character.

You should do following steps:

1. Usually, the raw data is not informative enough, and it is needed to extract informative features from raw data to build a good model. In this regard, four features (Hold Time “HT”, Press-Press time “PPT”, Release-Release Time “RRT”, Release-Press time “RPT”) are introduced and the definition of each of them are described in Fig 1.
2. For each row in Train_keystroke.csv, You should generate these features for each two consecutive keys.
3. After completing previous step, you should calculate mean and standard deviation for each feature per row. As a result, you should have 8 features (4 mean and 4 standard deviation) per row.
4. By using 8 generated features and UserID as class, build and train three ML models (1. SVM 2. Random Forest 3. XGBoost). (No need for feature selection, cross validation or parameters tuning, you should just train models on all data based on default parameters of each classifier)
5. Store the three trained models in your local device.

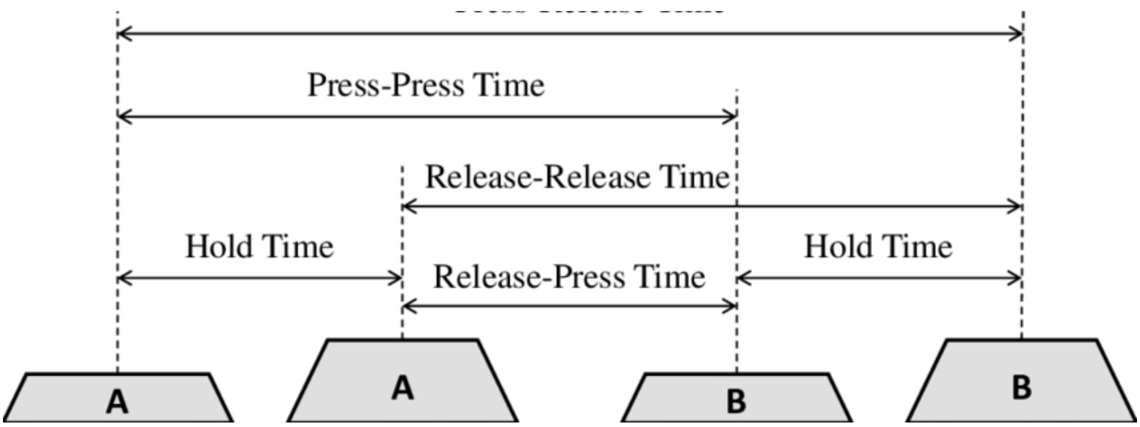

Fig 1. The definition of Hold time, Press-Press Time, Release-Release time, and Release-Press Time.

Part 2- Deploying Model

You are asked to deploy these three models as a single API to use them in real time. You should serve the model as a Rest API. This API gets inputs as json format and return the prediction result. The input for API should contains the model type (SVM, RF, XGB) and mean and std of HT, RPT, PPT and RRT. The sample format of input is as follow:

```
{
        “Model”: “RF”,
        "HT": {
                "Mean": 48.43,
                “STD”: 23.34
                },
        "PPT": {
                "Mean": 120.43,
                “STD”: 37.41
                },
        "RRT": {
                "Mean": 124.43,
                “STD”: 45.34
                },
        "RPT": {
                "Mean": 132.56,
                “STD”: 47.12
                }
}
```

When The API is called with propriate inputs, it should load the related model (based on “Model” value in input) and predict the UserID and return it.

For model deployment you can choose one of these platforms:

1. **AWS Platform**
   Consider AWS cloud for model deployment. You should provide your complete solution and are free to use any services in AWS.

2. **Microsoft Azure Platform**
   Provide your complete solution based on Microsoft Azure cloud.

Deliverable: We ask you the following deliverables:

1. A complete Jupyter notebook file with enough description for part 1 (Model Building)
2. For part 2 (Model Deployment) you should provide following items for one of platform (AWS, Azure)
   a. A fully comprehensive diagram showing various used modules and services with their internal relationship.
   b. A detailed step by step instruction guide on how to develop the deployment pipeline and serve model API.
   c. In case of any required script, you should provide the script with enough code comments in Python.

If you need any clarifications (e.g., definition), please do not hesitate to contact us on jobs@threatfabric.com.

You have maximum 10 days for this assignment, starting day after you receive this document as well as data.
Please send your solution to jobs@threatfabric.com.

We wish you a lot of fun with this challenge!