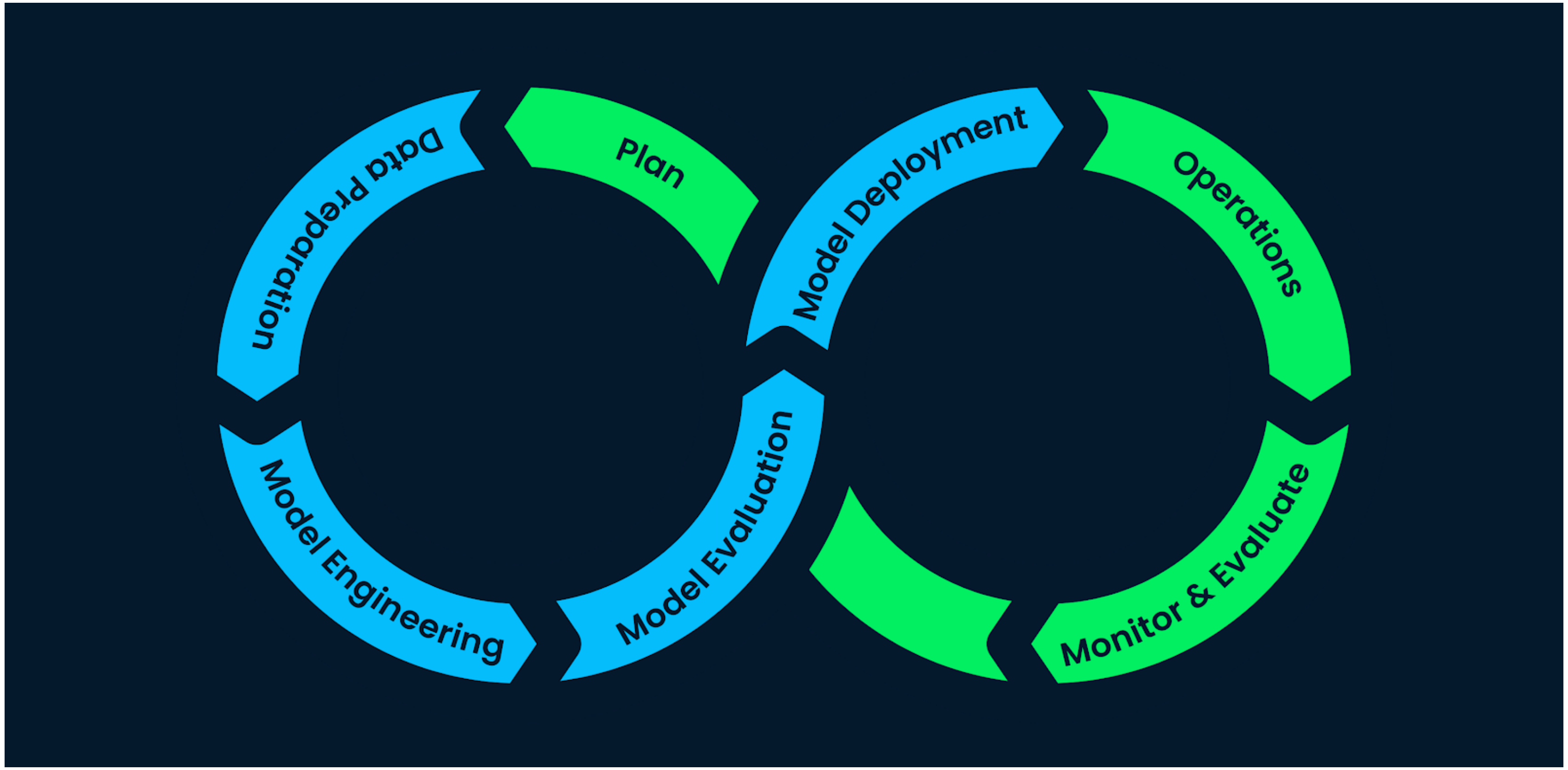
Machine Learning Diploma

Session : MLOps

Agenda

- ML life Cycle
- Machine Learning OPS
- Model Deployment
- Flask
- Hotel Reservation Prediction
- Deployment Model Using Flask
- Small Chatbot
- Deployment Model Using Flask

Machine Learning Life Cycle



Machine Learning Life Cycle

The 6 steps in a standard machine learning life cycle:

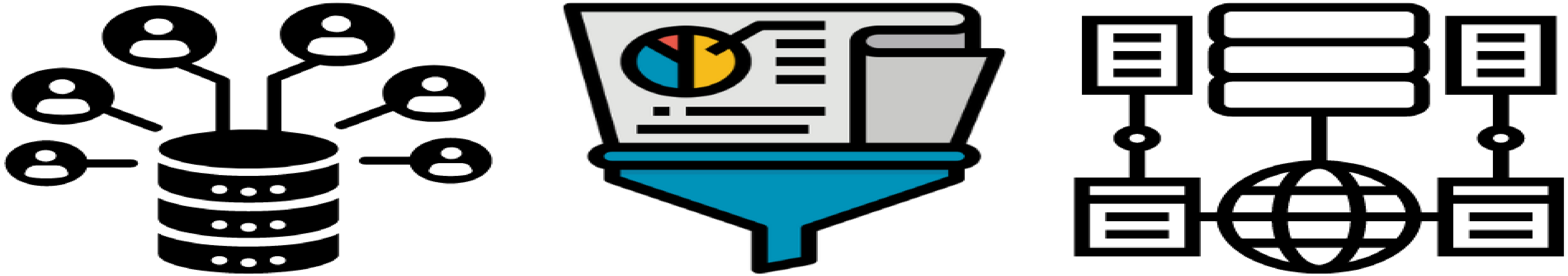
1. Planning
2. Data Preparation
3. Model Engineering
4. Model Evaluation
5. Model Deployment
6. Monitoring and Maintenance

1. Planning

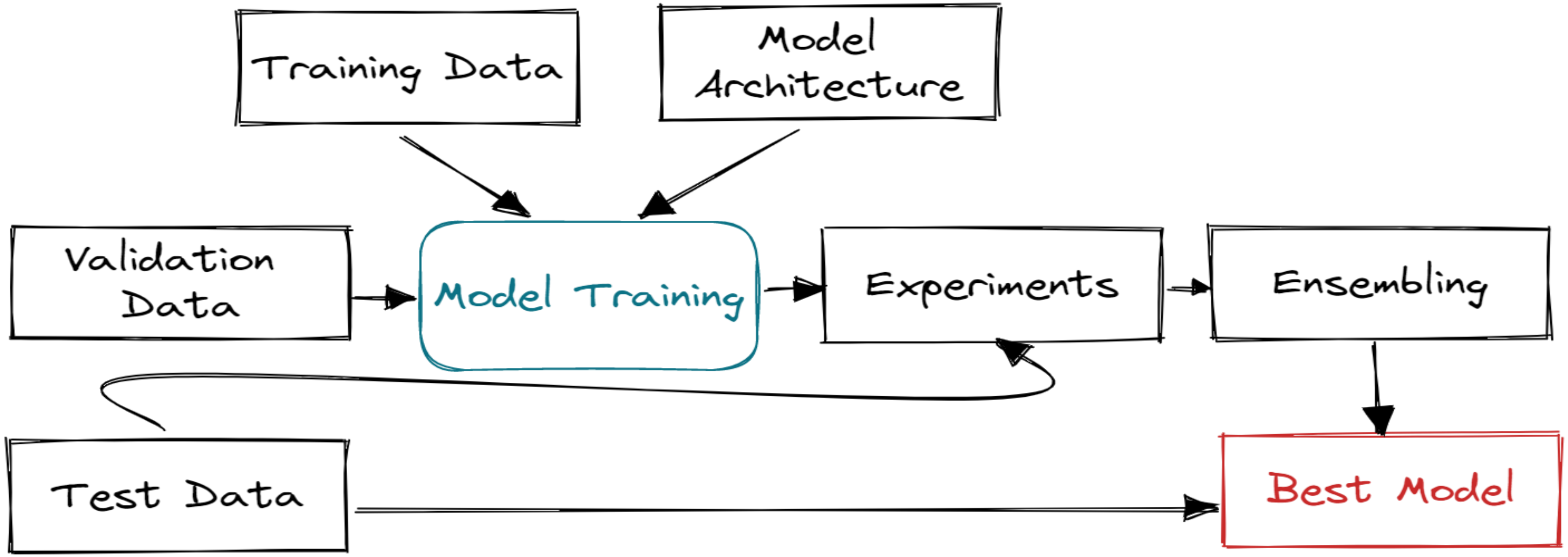
The planning phase involves assessing the scope, success metric, and feasibility of the ML application. You need to understand the business and how to use machine learning to improve the current process. For example: do we require machine learning? Can we achieve similar requests with simple programming?

2. Data Preparation

Data collection and labeling: we will collect the data by gathering the internal data, open-source, buying it from the vendors, or generating synthetic data.
Data Cleaning:, we will clean the data by imputing missing values, analyzing wrong-labeled data, removing outliers, and reducing the noise.
Data processing: The data processing stage involves feature selection, dealing with imbalanced classes, feature engineering, data augmentation, and normalizing and scaling the data.



3. Model Engineering



1. Define model
2. Split Data
3. Tune hyper parameter
4. Train model

4. Model Evaluation

- We will first test our model on a test dataset and make sure we involve subject matter experts to identify the error in the predictions.
- Furthermore, we will test our model for robustness on random and real-world data. Making sure that the model inferences fast enough to bring the value.
- Finally, we will compare the results with the planned success metrics and decide on whether to deploy the model or not. In this phase, every process is recorded and versioned to maintain quality and reproducibility.

5. Model Deployment

- Deploy machine learning model on the cloud and local server, web browser, package as software, and edge device. After that, you can use API, web app, plugins, or dashboard to access the predictions.

6. Monitoring and Maintenance

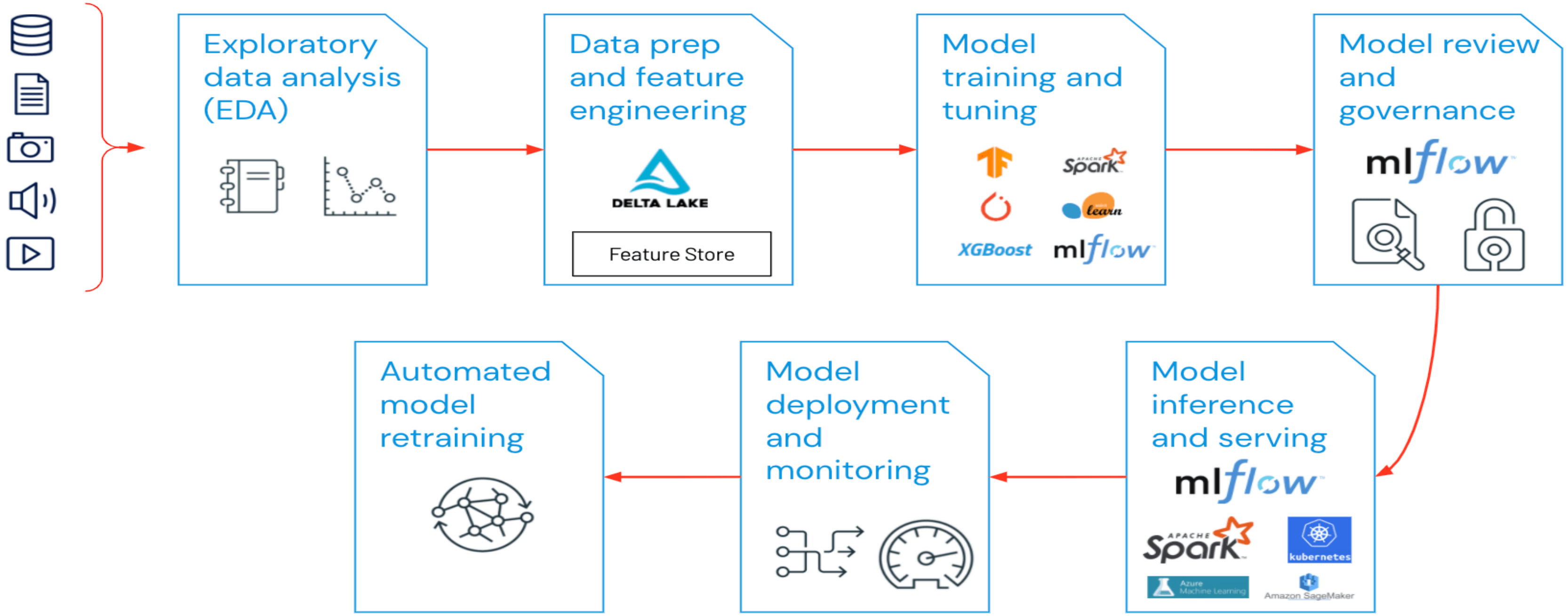
1. After deploying the model to production we need to constantly monitor and improve the system. We will be monitoring model metrics, hardware and software performance, and customer satisfaction.

2. The monitoring is done completely automatically, and the professionals are notified about the anomalies, reduced model and system performance, and bad customer reviews.

3. After we get a reduced performance alert, we will assess the issues and try to train the model on new data or make changes to model architectures. It is a continuous process.

MLOps

MLOps is a useful approach for the creation and quality of machine learning and AI solutions. By adopting an MLOps approach, data scientists and machine learning engineers can collaborate and increase the pace of model development and production, by implementing continuous integration and deployment (CI/CD) practices with proper monitoring, validation, and governance of ML models.

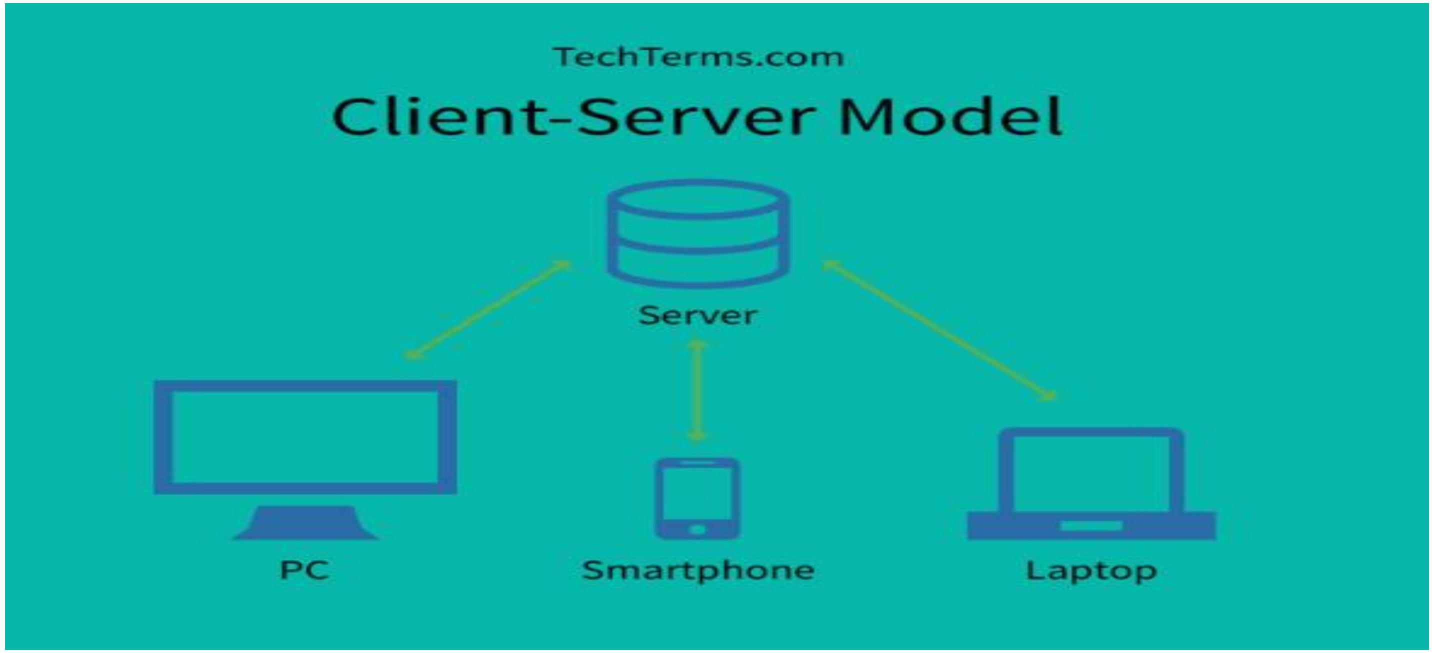

Exploratory data analysis (EDA)
Data prep and feature engineering
DELTA LAKE
Feature Store
Model training and tuning
Spark
learn
XGBoost
mlflow
Model review and governance
mlflow
Model inference and serving
mlflow
Spark
kubernetes
Azure Machine Learning
Amazon SageMaker
Model deployment and monitoring
Automated model retraining

Model Deployment

Two types :

1. Client Side Deployment
2. Server Side Deployment

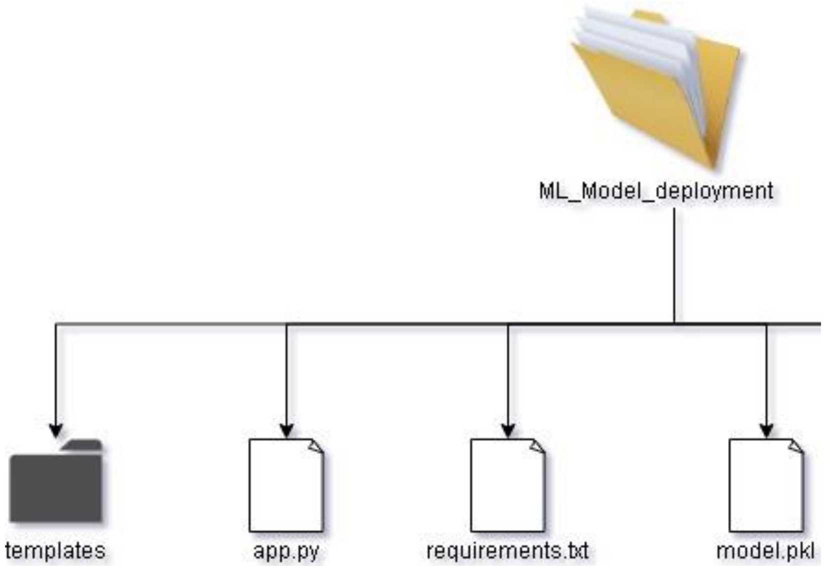
Server Side Deployment:



Flask

Flask is a **web application framework written in python**, in simple terms it helps end users interact with your python code (in this case our ML models) directly from their web browser without needing any libraries code files, etc.

Flask enables you to create web applications very easily, hence enabling you to focus your energy more on other important parts of a ML lifecycle like EDA, feature engineering, etc. Here in this blog I will give you a walkthrough on how to build a simple web application out of your ML Model and deploying it eventually.



Flask Directory Structure

•**templates** :- This folder contains the html files *(index.html, predict.html)* that would be used by our main file *(app.py)* to generate the front end of our application

•**static** :- This folder contains the css files.

app.py :- This is the main application file, where all our code resides, and it binds everything together.

requirements.txt :- This file contains all the dependencies/libraries that would be used in the project (whenever a virtual environment is created it can use this requirements file directly to download all the dependencies you need not to install all the libraries manually, you just need to put all of them in this file)

model.pkl :- This is our saved model.

End-to-end machine learning project: Hotel Reservation

THANK YOU!

AMIT