Phase-5-Machine Learning Model Deployment with IBM Cloud Watson Studio

Abstract

Machine learning model deployment is the process of integrating a trained machine learning model into a real-world system or application to automatically generate predictions or perform specific tasks. This project documentation describes the deployment of a machine learning model for a predictive use case, including the dataset selection, model training, deployment process, and integration steps. It also explains how the deployed model can be accessed and utilized for real-time predictions.

Phase 5: Project Documentation & Submission

Documentation

The project documentation should outline the project's objective, design thinking process, and development phases. It should also describe the predictive use case, dataset selection, model training, deployment process, and integration steps.

Objective

The objective of the project is to deploy a machine learning model to predict a target variable of interest.

Design Thinking Process

The design thinking process for this project involved the following steps:

1**. Empathize:** Understand the needs of the end users and the problem that the machine learning model will solve.
2. **Define**:Clearly define the problem statement and the scope of the project.
3. **Ideate:** Brainstorm potential solutions to the problem, including different machine learning algorithms and dataset selection strategies.
4. **Prototype:**Implement a prototype of the machine learning model and evaluate its performance on a held-out test set.
5. **Test:**Deploy the machine learning model to a production environment and monitor its performance in real time.

Development Phases

The development phases for this project were as follows:

1. **Dataset selection:**Select a dataset that is relevant to the predictive use case and that contains the necessary features to train the machine learning model.
2. **Data preprocessing:** Clean and prepare the dataset for machine learning by handling missing values, outliers, and other data quality issues.
3**. Feature engineering:**Create new features from the existing data or transform the data in a way that improves the performance of the machine learning model.
4. **Model training:**Train a machine learning model on the preprocessed dataset and evaluate its performance on a held-out test set.
5. **Model deployment:**Deploy the trained machine learning model to a production environment so that it can be used to make predictions on new data.

Predictive Use Case

The predictive use case for this project is to predict the likelihood of customer churn. The dataset contains historical data on customer behavior, such as purchase history, support tickets, and engagement with marketing campaigns. The machine learning model will be trained on this data to learn the patterns that are associated with customer churn. Once the model is deployed, it can be used to predict the likelihood of churn for new customers, which can help businesses to identify and retain at-risk customers.

Dataset Selection

The dataset for this project was selected based on the following criteria:

* **Relevance to the predictive use case:** The dataset must contain the necessary features to predict the target variable of interest (customer churn).
* **Completeness:** The dataset must contain a sufficient number of samples and features to train the machine learning model.
* **Quality:** The dataset must be clean and free from errors.

Model Training

The following machine learning algorithms were trained on the selected dataset:

* Logistic regression
* Random forest
* Gradient boosted trees

The model with the best performance on the held-out test set was selected for deployment.

Deployment Process

The machine learning model was deployed to a web service using the following steps:

1. Create a Dockerfile to containerize the machine learning model and its dependencies.
2. Build the Docker image and push it to a Docker registry.
3. Deploy the Docker image to a cloud platform, such as Amazon Elastic Container Service (ECS) or Kubernetes.
4. Create an API endpoint to expose the machine learning model to end users.

Integration Steps

The machine learning model was integrated with the existing customer relationship management (CRM) system using the following steps:

1. Create a new API endpoint in the CRM system to expose the machine learning model.
2. Develop a custom integration script to call the machine learning model API from the CRM system.
3. Configure the CRM system to use the custom integration script to predict the likelihood of churn for new customers.

Accessing and Utilizing the Deployed Model for Real-Time Predictions

The deployed machine learning model can be accessed and utilized for real-time predictions by making API requests to the exposed API endpoint. The API request should include the