ARCHITECTURE LEVEL DESIGN

Bike Sharing Demand Prediction

09/11/2024

Ritik Patel

Content

1 Introduction

1.1 What is Architecture-Level Design Document?

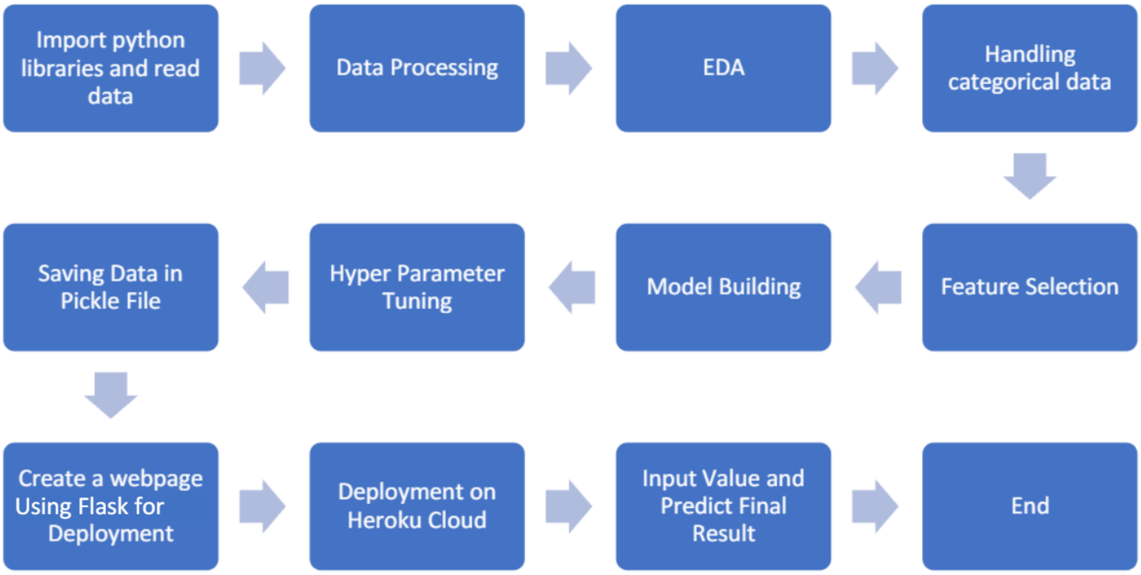
The Architecture-Level Design Document provides an overview of the architectural design and components of the bike sharing demand prediction project. It outlines the structure and functionality of the system at a high level.

1.2 Scope

The scope of this document is to describe the architecture and design considerations for the bike sharing demand prediction project. It covers the data gathering, data preprocessing, model building, data validation, rendering result, and deployment phases of the project.

2 Architecture

The architecture of the bike sharing demand prediction project follows a modular and layered approach. It consists of several components such that work together to predict the demand for bike sharing based on various factors. They are


Import python libraries and read data
Data Processing
EDA
Handling categorical data
Feature Selection
Model Building
Hyper Parameter Tuning
Saving Data in Pickle File
Create a webpage Using Flask for Deployment
Deployment on Heroku Cloud
Input Value and Predict Final Result
End

3 Architecture Description

3.1 Data Gathering

The data gathering phase involves collecting relevant data for the bike sharing demand prediction. This may include historical bike sharing data, weather data, holiday information, and other relevant factors that can influence the demand which can be gathered from Kaggle using Kaggle API "kaggle competitions download -c bike-sharing-demand".

3.2 Data Description

In the data description phase, the collected data is analysed and described using google colab. Following are the variables present in the dataset :-

- **datetime**- hourly date + timestamp .
- **season** - 1 = spring, 2 = summer, 3 = fall, 4 = winter .
- **holiday** - whether the day is considered a holiday.
- **workingday** - whether the day is neither a weekend nor holiday.
- **weather**
    1. Clear, Few clouds, Partly cloudy, Partly cloudy.
    2. Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    3. Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    4. Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- **temp** - temperature in Celsius
- **atemp** - " feels like" temperature in Celsius
- **humidity** - relative humidity
- **windspeed** - wind speed
- **casual**- number of non-registered user rentals initiated
- **registered** - number of registered user rentals initiated
- **count** - number of total rentals

These attributes such as datetime, season, weather, temperature, humidity, and other variables are examined to understand their significance and impact on the demand.

3.3 Tool Used

Various tools and technologies are utilized in the bike sharing demand prediction project. They are

- Python programming language.
- Libraries such as Pandas and Scikit-learn for data manipulation and modelling .
- VSCODE is employed as IDE.
- Visualization tools for result analysis such as seaborne and matplotlib.
- Heroku/Streamlit is employed for deployment

3.4 Data Pre-processing

Data pre-processing involves cleaning, transforming, and preparing the data for model building.

Steps performed in pre-processing are:

- Checked for null values /missing value.
- Checked for duplicates in the dataframe.
- Checked for outliers in the dataframe.
- Scaling is performed using standard scaler.

3.5 Model Building

In the model building phase, various machine learning models are trained using the pre-processed data. Models such as

1) Linear Regression
2) Random Forest
3) Extra Trees,
4) LightGBM are used to predict the bike sharing demand based on the input features.

3.6 Hyperparameter Tuning

Hyperparameter tuning is performed and best model which is then saved using pickle file.

3.7 Data from User

The system allows the user to input specific data, such as season,holiday,working_day etc to get predictions for future bike sharing demand. This input is processed and utilized by the trained models to generate accurate predictions.

3.8 Data Validation

Data validation is performed to assess the accuracy and reliability of the predicted demand. This involves taking data from the user which is processed in app.py file. The validated data is then sent for the prediction.

3.9 Rendering Result

The results of the bike sharing demand prediction are rendered to correctly predict bike sharing demand based on the input provided by the user.

4 Deployment

Once the model is trained and validated, it is deployed to a production environment. This allows the system to handle real-time prediction requests and provide accurate bike sharing demand forecasts.