BIKE SHARING DEMAND PREDICTION

Detailed Project Report

09/11/2024

Ritik Patel

Contents

1 Introduction:

The high-level design document provides an overview of the project, which aims to develop a bike sharing demand prediction system. The system will utilize machine learning techniques to forecast the number of bikes required at different hours of the day for a stable supply of rental bikes. This document outlines the key aspects of the project's design and implementation.

1.1 What is High-Level Design Document?

A high-level design document is a blueprint that outlines the overall structure and components of a project. It provides a conceptual understanding of the system architecture, major functionalities, and interactions between different modules. It serves as a guide for the development team and stakeholders to understand the project at a high level.

1.2 Scope:

The scope of this project includes developing a bike sharing demand prediction system that can accurately forecast the required number of bikes for each hour of the day. The system will take into account various factors such as season, weather conditions, and special events to make accurate predictions. The target audience for this system includes bike rental companies, urban city planners, and bike enthusiasts.

2 Description:

2.1 Problem Perspective:

The problem addressed by this project is the need for a stable supply of rental bikes in urban cities. It is important to accurately predict the demand for bikes at different hours to ensure availability and reduce waiting time for users. By accurately forecasting the demand, bike rental companies can optimize their inventory and improve customer satisfaction.

2.2 Problem Statement:

The problem statement for this project is to develop a bike sharing demand prediction system that can predict the number of bikes required at each hour of the day based on various factors such as season, weather conditions, and historical usage patterns. The system should provide accurate predictions to ensure a stable supply of rental bikes.

2.3 Proposed Solution:

The proposed solution is to build a machine learning model that can analyse historical bike usage data and other relevant factors to predict the demand for bikes at different hours. The model will be trained using supervised learning techniques, and various features such as season, weather conditions, and time of day will be considered to improve prediction accuracy.

2.4 Solution Improvements:

To enhance the solution, we will explore techniques such as feature engineering to extract more meaningful information from the data. Additionally, ensemble methods or advanced machine learning algorithms can be employed to further improve prediction accuracy. Regular model retraining and updating with new data will also be considered to adapt to changing demand patterns.

2.5 Technical Requirements:

The technical requirements for this project include a programming language (e.g., Python), machine learning libraries (e.g., scikit-learn, TensorFlow), data storage and processing infrastructure, and a web application framework for visualization and result rendering.

2.6 Data Requirements:

The project requires historical bike usage data, which should include attributes such as date and time, season, weather conditions, and the number of bikes

rented. Additional data sources may include weather data, calendar events, and demographic information to enrich the prediction model.

2.7 Tool Used:

The project will utilize Python as the primary programming language for data analysis, modeling, and visualization. Machine learning libraries such as scikit.

2.8 Data Gathering:

The data gathering process involves collecting historical bike usage data from rental bike companies or relevant sources. This data can be obtained through APIs, database queries, or direct collaboration with bike rental companies. Additionally, weather data can be sourced from weather APIs or meteorological services. Calendar events and demographic information can be obtained from public sources or relevant databases.

2.9 Data Description:

The collected data will include attributes such as the

- instant (timestamp of data)
- dteday (date)
- season (categorical variable representing the season)
- yr (year)
- mnth (month)
- hr (hour)
- holiday (binary variable indicating whether it is a holiday)
- weekday (categorical variable representing the day of the week)
- workingday (binary variable indicating whether it is a working day)
- weathersit (categorical variable representing weather conditions)
- temp (temperature)
- atemp (adjusted temperature)
- hum (humidity)
- windspeed (wind speed)
- casual (number of casual users)

- registered (number of registered users)
- cnt (total number of bike rentals).

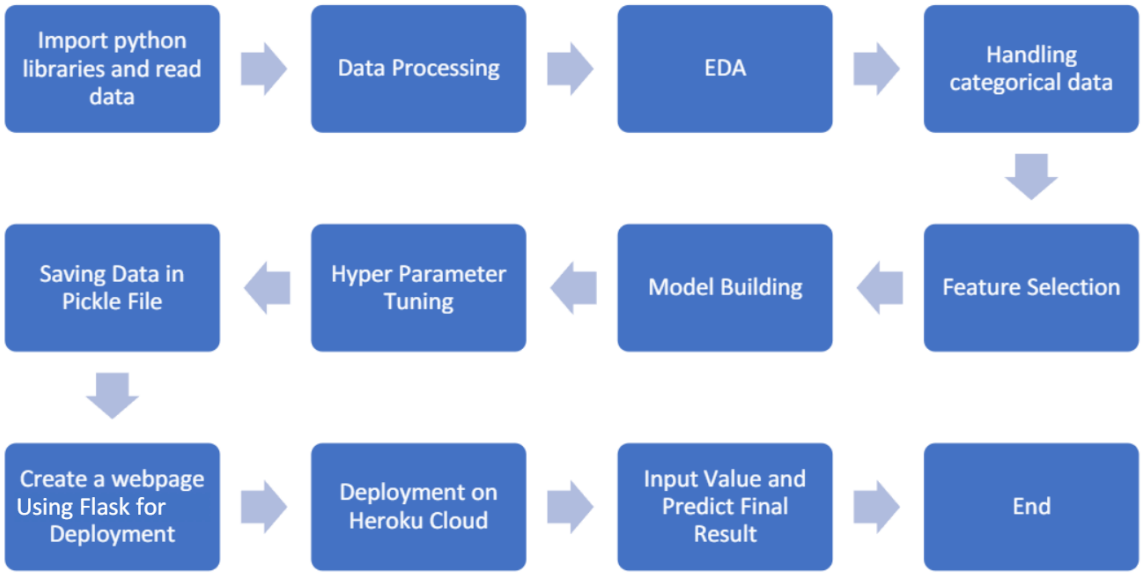
3 Data Pre-Processing:

Before modeling, the collected data will undergo pre-processing steps to ensure its quality and suitability for analysis. This include

- Handling missing values
- Data normalization or scaling
- Encoding categorical variables
- Removing outliers.
- Feature engineering techniques may also be applied to create new features or extract relevant information from existing ones.

4 Design Flow:

The design flow of the project involves several stages which are



4.1 Modelling:

In the modelling stage, various machine learning algorithms will be explored and evaluated for their suitability in predicting bike demand. Techniques such

as regression, time series analysis, or ensemble methods may be employed to develop accurate prediction models.

4.2 Modelling Process:

The modelling process includes

- data splitting into training and testing sets
- feature selection or extraction
- model training using the training set, and model evaluation using the testing set.
- Iterative refinement of the models may be performed by adjusting hyperparameters or trying different algorithms to optimize performance.

4.3 Deployment Process:

Once the prediction model is developed and evaluated, it will be deployed in a production environment. This may involve building a web application or an API that can take input parameters (such as date, time, weather conditions) and provide the predicted bike demand as output. The deployment process will also consider scalability, performance optimization, and security aspects.

5 Data from User:

The system may allow users to input certain parameters, such as date and time, to obtain predictions for specific time periods. User interaction with the system will be designed to be user-friendly and intuitive.

6 Data Validation:

Data validation techniques will be implemented to ensure the accuracy and reliability of the input data. This may involve checking for outliers, verifying the completeness and consistency of the data, and validating the inputs against predefined criteria.

7 Rendering Result:

The system will provide the predicted bike demand as a result, which can be displayed through visualizations such as charts, graphs, or tables. The results can be rendered on a web interface or presented in a user-friendly format for easy interpretation.

8 Conclusion:

- The model built using the LightGBM algorithm is the most accurate one.
- Decision tree-based algorithms are more accurate than linear regression-based algorithms.
- If not tuned properly, the Random forest Regressor can overfit the training data, which can lead to poor generalization on the test data.
- Hyperparameter tuning time is pretty high for the larger dataset in Random forest and XG Boost.
- There is a trade-off between model accuracy and time consumed for model creation. So according to the desired requirement of the company model should be selected i.e if high accuracy is desired then XGboost is the best model else if one is dealing with a huge dataset and time is a constrain along with model interpretability is desired then the decision tree model can be selected.

9 .Questions & Answers:

Q1) What's the source of data?

The data for training is provided by the client in multiple batches and each batch contain multiple files.

Q 2) What was the type of data?

The data was the combination of umerical and Categorical values.

Q 3) What's the complete flow you followed in this Project? Refer Page no 6 for better Understanding.

Q 4) How logs are managed?

Different logs as per the steps that we follow in validation and modelling like File validation log, Data Insertion, Model Training log, prediction log etc. are created.

Q 5) What techniques were you using for data pre-processing?

• Removing unwanted attributes

• Visualizing relation of independent variables with each other and output variables using box-plot.

• Removing outliers

• Cleaning data and imputing if null values are present.

• Converting categorical data into numeric values.

Q 6) How training was done or what models were used?

• Before dividing the data in training and validation set, pre-processing was performed over the data set and then final dataset was created.

• As per the dataset training and validation data were divided.

• Algorithms like Linear regression, Extra Decision Tree, Random Forest, LightGBM were used and based on the R2 score, final model created which is then saved for future prediction.

Q 7) What are the different stages of deployment?

• First, the scripts are stored on GitHub as a storage interface.

• The model is first tested in the local environment.

• After successful testing, it is deployed on Heroku/Streamlit.