RESTAURANT RATING PREDICTION

ARCHITECTURE DESIGN

BY
Harshitha AV
Spoorthi Chandra.P

ABSTRACT

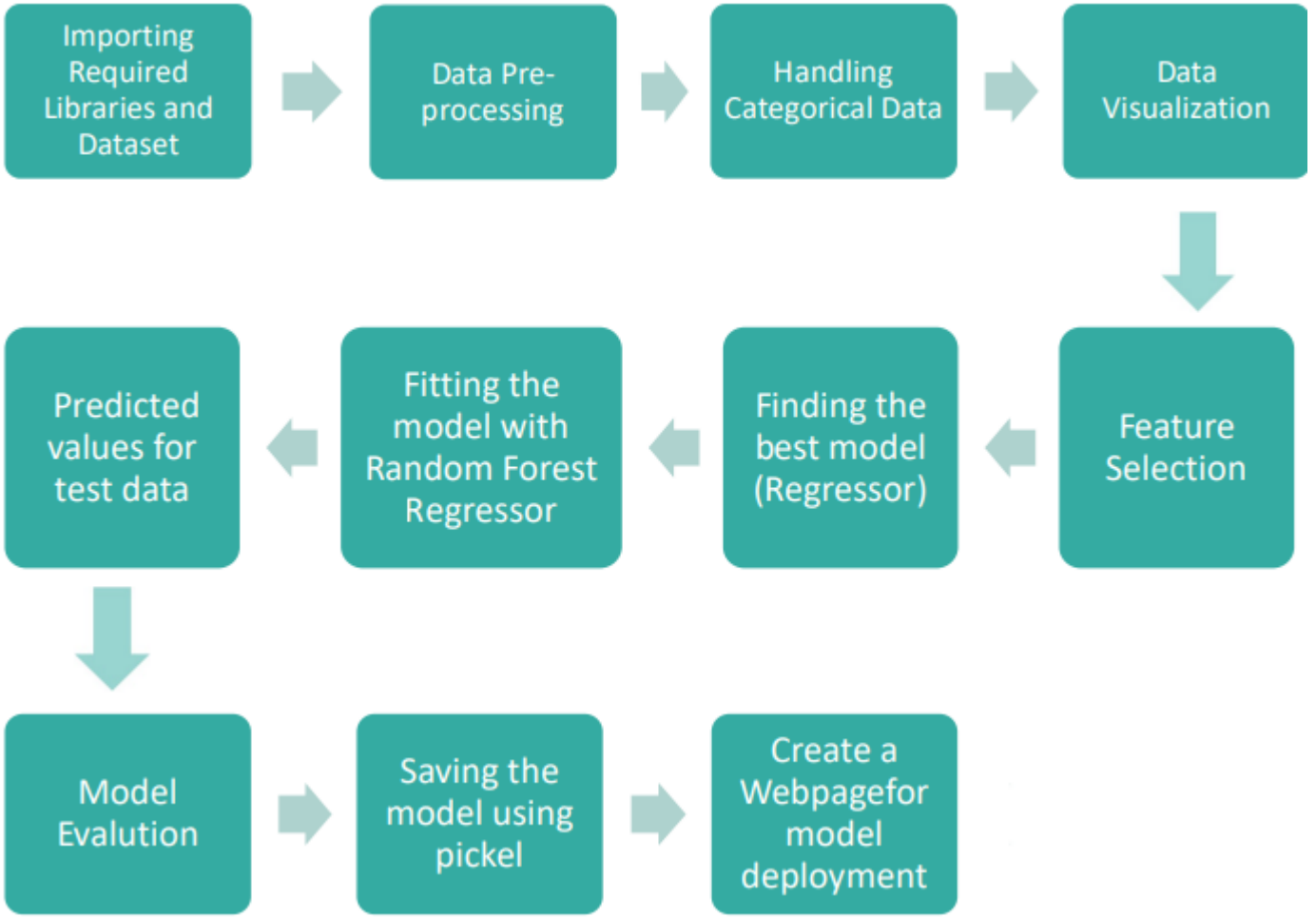
The basic idea of analyzing the Zomato dataset is to get a fair idea about the factors affecting the establishment of different types of restaurants at different places in Bengaluru, aggregate rating of each restaurant, Bengaluru being one such city has more than 50,000 restaurants with restaurants serving dishes from all over the world. With each day new restaurants opening the industry hasn't been saturated yet and the demand is increasing day by day. Bengaluru being an IT capital of India, most of the people here are dependent mainly on the restaurant food as they don’t have time to cook for themselves. With such an overwhelming demand for new restaurants, it has become important to study the ratings of restaurants.

INTRODUCTION

What this Architecture Design Document ?

The main objective of the Architecture design documentation is to provide the internal logic understanding of the flight fare prediction code. The Architecture design documentation is designed in such a way that the programmer can directly code after reading each module description in the documentation.

ARCHITECTURE



Architecture Design

Data Collection

The data for this project is collected from the Kaggle Dataset, the URL for the dataset is https://www.kaggle.com/datasets/himanshupoddar/zomato-bangalore-restaurants

Data Description

The dataset contains 17 variables all of which were scrapped from the Zomato website. The dataset contains details of more than 50,000 restaurants in Bengaluru in each of its neighborhood. The total size of dataset is approximately 547 MB.

- url: contains the url of the restaurant in the zomato website
- address: contains the address of the restaurant in Bengaluru
- name: contains the name of the restaurant
- online_order: whether online ordering is available in the restaurant or not
- book_table: table book option available or not
- rate: contains the overall rating of the restaurant out of 5

- votes: contains total number of rating for the restaurant as of the above mentioned date.
- phone: contains the phone number of the restaurant.
- location: contains the neighborhood in which the restaurant is located.
- rest_type: restaurant type.
- dish_liked: dishes people liked in the restaurant.
- cuisines: food styles, separated by comma.
- approx_cost(for two people): contains the approximate cost for meal for two people.
- reviews_list: list of tuples containing reviews for the restaurant, each tuple.
- menu_item: contains list of menus available in the restaurant.
- listed_in(type): type of meal.
- listed_in(city): contains the neighborhood in which the restaurant is listed.

Importing dataset and libraries:

We have imported certain libraries such as numpy,panda,matplot and many more.
The ZOMATO dataset is imported in the form of csv.

Data Preprocessing

Checked for info of the Dataset, to verify the correct datatype of the Columns.
Checked for Null values, because the null values can affect the accuracy of the model.
Converted all the illegal values into legal values.
Checking the distribution of the columns to interpret its importance.
Prepared the relevant data from the dataset.
Now, the info is prepared to train a Machine Learning Model.

Modelling Creation

After preprocessing the data, we visualize our data to gain insights and then these insights are randomly spread and split into two parts, train and test data. After splitting the data, we use Random Forest Regressor to model our data to predict the Restaurant Rating.

UI Integration

Both CSS and HTML files are being created and are being integrated with the created machine learning model. All the required files are then integrated to the app.py file and tested locally.

Data from User

The data from the user is retrieved from the created HTML web page.

Data Validation

The data provided by the user is then being processed by app.py file and validated. The validated data is then sent to the prepared model for the prediction. Importing Required Libraries and Dataset Data Preprocessing Handling Categorical Data Data Visualization Predicted values for test data Fitting the model with Random Forest Regressor Finding the best model (Regressor) Feature Selection Model Evalution Saving the model using pickel Create a Webpage for model deployment Deployment.

Rendering the Results

The data sent for the prediction is then rendered to the web page.

Deployment

The tested model is then deployed. So, users can access the project from any internet devices.