

**Zomato Restaurant Rating Prediction**

Project Architecture

Domain: Machine Learning

Creator: Dipesh Sharma

Date: 25.07.2024

Dipesh Sharma

+91 9548792372

[ds4486487@gmail.com](mailto:talaripraveenkumar53@gmail.com)

**Architecture**

**Data Preparation**

**Model Development**

**Deployment**

**Deployment Process**

**Architecture Description**

**Data Preparation**

Data Description

I was always fascinated by the food culture of Bengaluru. Restaurants from all over the world can be found here in Bengaluru. From United States to Japan, Russia to Antarctica, you get all type of cuisines here. Delivery, Dine-out, Pubs, Bars, Drinks, Buffet, Desserts, you name it and Bengaluru has it. Bengaluru is best place for foodies. The number of restaurants are increasing day by day. Currently which stands at approximately 12,000 restaurants. With such a high number of restaurants. This industry hasn't been saturated yet. And new restaurants are opening every day. However, it has become difficult for them to compete with already established restaurants. The key issues that continue to pose a challenge to them include high real estate costs, rising food costs, shortage of quality manpower, fragmented supply chain and over-licensing

Data Preprocessing

In data preprocessing step, we check if there missing data, duplicate values, and datatypes of each feature. In our dataset, there was not any null and duplicate values but datatype of “data\_time” column was “object”; thus, it was converted to “datetime64”.

Exploratory Data Analysis

This step includes bivariate and univariate analysis of features. Checking outliers using boxplots, and outlier treatment is carried out as well. Distribution of numerical values is plotted to see to what extent our data is skewed.

Feature Engineering

In this part, compute aggregate statistics from existing features to provide a summary view of the data. For instance, calculate features like average rating per cuisine, average price range per location, or the number of restaurants in a specific category within a certain radius. If available, include features that capture user interactions with the Zomato platform. This might include features such as the number of reviews a user has submitted, average rating given by the user, or the total number of votes received for their reviews.

**Model Development**

Model implementation

After train and test splitting, pipeline containing Standard Scaler and Ordinal Encoder was fitted to several models such as AdaBoost Regressor, Gradient Boosting Regressor, RandomForest Regressor, CatBoost Regressor, XGB Regressor. Their R2 score were obtained. The highest score is acquired from the RandomForest Model.

Hyper-parameter Tuning

The best model is chosen, and Grid Search with Cross Validation is applied on that model to get the best parameters. Those parameters is then used on the model to get better result.

Model Evaluation

Test dataset is used to evaluate the model. 20% of dataset was separated for testing. Predicted results of the model are compared with the actual data to check the amount of error.

**Deployment**

Designing UI with Anvil

For this project, a user interface is built on Anvil. It is a web application that helps us to create applications for projects. It is a free Python-based drag-and-drop web app builder.

Designing a server

A server should be created to run the UI application continuously. Flask server is built and it is linked with Anvil uplink that connects Anvil UI with our server.

Code deployment on cloud

The codes for this machine learning model should be deployed to the cloud, so that when data is entered into the application, our code runs and a user gets the result online.

**Deployment Process**

In this stage, we establish a server using Flask that runs the uplink code (server code) in parallel before developing the UI using Anvil and connecting with our code, where our model is executing, via an uplink. We will post the hole after execution or asynchronous execution. Git and Github are used to code in the Heroku cloud. Then, we'll configure a cron job to maintain the server and server code in operation indefinitely.