RRP (RESTRAUNT RATING PREDICTION)
Regression Problem

Revision Number: 1.0
Last date of revision: 24/03/2023

Document Version Control

Date Issued	Version	Description	Author
24/03/2023	1	Initial HLD — V1.0	Rahul Chatterjee

Contents

Abstract

Zomato is one of the largest online restaurant discovery and food ordering platforms. The Zomato dataset contains information on thousands of restaurants, including their ratings, menus, and customer reviews. The objective of this project is to develop a machine learning model that can predict the ratings of restaurants on the Zomato dataset based on various factors such as the quality of food, service, ambiance, and location.

The proposed solution utilizes supervised machine learning algorithms such as linear regression, decision trees, and random forests to predict the ratings of restaurants on the Zomato dataset. The data is preprocessed to ensure that the input data is appropriate for the selected algorithms. Feature selection and feature engineering are performed to select the most relevant features that influence the ratings of restaurants.

The model is trained on a large dataset and validated using cross-validation techniques to evaluate its performance. The results show that the model can accurately predict the ratings of restaurants on the Zomato dataset with a high degree of accuracy.

The developed model can help restaurant owners to identify areas for improvement and make data-driven decisions to enhance customer satisfaction. The model can also assist customers in selecting restaurants based on their preferences and rating.

In conclusion, this project demonstrates the potential of machine learning algorithms in predicting restaurant ratings on the Zomato dataset accurately. Future work could involve exploring other machine learning techniques and integrating real-time data to enhance the accuracy of the model.

1. Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
    - Security
    - Reliability
    - Maintainability
    - Portability
    - Reusability
    - Application compatibility
    - Resource utilization
    - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

1.3 Definitions

TermRRP	*Description*
Database	Restaurant Rating Prediction
IDE	Collection of all the information monitored by this system
AWS	Integrated Development Environment
	Amazon Web Services

2. General Description

2.1 Product Perspective

The RRP is a regression model which will help us to predict the Rating of restaurant based on the given parameters.

2.2 Problem statement

Zomato, the online food app, has reached out to us to help them to predict how good or bad a restaurant will turn out in the future. So that, they can take a decision to include the restaurant in their app or remove it.
They have shared the data of 9551 restaurants from all over the world which are currently present in the Zomato App. It contains the details about the restaurants and what rating it achieved finally.
Our task is to create a machine learning model which can predict the Rating of a restaurant based on its characteristics.

2.3 PROPOSED SOLUTION

1. Data Ingestion: Collect data from Zomato dataset. The data should include information on restaurant attributes, such as location, cuisine, price range, menu items, and customer ratings.
2. Data Preprocessing: Clean the data by removing irrelevant or missing values, and performing feature engineering to extract useful features from the data. Also, normalize or scale the data to ensure that all features have equal importance.
3. Feature Selection: Select the most relevant features that influence the restaurant ratings. This can be done by using statistical methods, such as correlation or feature importance analysis.
4. Model Training: Train the machine learning model using various algorithms, such as linear regression, decision trees, or random forests. The data should be split into training and testing datasets to evaluate the model's performance.
5. Model Evaluation: Evaluate the model's performance using metrics such as accuracy, precision, recall, or F1-score. The model should be tested on the testing dataset to ensure its generalization ability.
6. Model Deployment: Deploy the model for use in real-world scenarios, such as recommending restaurants to customers based on their preferences or assisting restaurant owners in making data-driven decisions.

Overall, the proposed solution involves collecting and preprocessing data, selecting relevant features, training the machine learning model, evaluating its performance, and deploying the model for real-world applications. The solution can assist restaurant owners in understanding their customers' preferences and making data-driven decisions to enhance customer satisfaction.

2.4 FURTHER IMPROVEMENTS

1. Incorporating Deep Learning: Deep learning algorithms such as neural networks can be used to capture complex relationships and patterns between different restaurant attributes and ratings. This could improve the accuracy of the model and enhance its generalization ability.
2. Fine-tuning Hyperparameters: Hyperparameters such as learning rate, number of hidden layers, and regularization can significantly impact the performance of the model. Fine-tuning these hyperparameters using techniques such as grid search or random search can help to optimize the model's performance.
3. Incorporating Time-series Analysis: Incorporating time-series analysis can help capture trends in restaurant ratings over time, and identify factors that influence ratings during specific time periods. This could help restaurant owners to make informed decisions on how to adjust their operations to improve ratings during specific time periods.
4. Incorporating Sentiment Analysis: Sentiment analysis can be used to analyze customer reviews and determine the sentiment of the review. This could be used to identify specific aspects of the restaurant that customers like or dislike, and inform restaurant owners on what areas to focus on for improvement.
5. Integration with Mobile Applications: Integration with mobile applications can help to improve the accessibility and usability of the model. This could allow customers to easily find restaurants that meet their preferences, and allow restaurant owners to monitor their ratings and make data-driven decisions on-the-go.

Overall, these improvements can enhance the accuracy, usability, and scalability of the solution, and enable restaurant owners to make data-driven decisions that improve customer satisfaction and business outcomes.

2.5 Technical Requirements

1. Programming Languages and Tools: The solution may require the use of programming languages such as Python or R for data processing, analysis, and modeling. Additionally, libraries and frameworks such as scikit-learn, TensorFlow, and Keras may be necessary for implementing various machine learning algorithms.
2. Data Storage and Retrieval: A database or data storage system such as MySQL, MongoDB, or Hadoop may be necessary to store and retrieve large amounts of data used for training and testing the machine learning model.
3. Cloud Computing: Cloud computing platforms such as AWS, GCP, or Azure may be required for processing and analyzing large datasets and implementing complex machine learning algorithms.
4. User Interface: A user interface such as a web application or mobile application may be necessary to provide an easy-to-use interface for restaurant owners and customers to interact with the model.
5. Data Security: Data security measures such as encryption, access control, and data anonymization may be required to ensure the privacy and security of customer and restaurant data.
6. Deployment: The solution may need to be deployed on a cloud-based or on-premise infrastructure, depending on the specific requirements of the project.

Overall, the technical requirements for implementing the solution would depend on the specific implementation details and the scale of the project. A thorough analysis of the requirements, available resources, and project scope would be necessary to determine the optimal technical approach.

2.6 Data Requirements

Data requirement completely depend on our problem statement.

1. Restaurant Data: Data on restaurants, such as location, cuisine, price range, menu items, opening hours, and other relevant attributes. This data can be obtained from restaurant websites, social media platforms, or online directories.
2. Customer Data: Data on customers, such as age, gender, income, location, and other relevant attributes. This data can be obtained from customer reviews, social media platforms, or online surveys.
3. Ratings Data: Data on restaurant ratings, including the rating score and the number of ratings. This data can be obtained from restaurant review websites such as Zomato, Yelp, or TripAdvisor.
4. Feature Data: Data on the features or characteristics of the restaurant that may influence its rating, such as ambiance, service, food quality, and price. This data can be obtained from customer reviews, restaurant websites, or other online sources.
5. Time-series Data: Time-series data on restaurant ratings over a specific period. This data can help to identify trends and patterns in restaurant ratings over time, and inform decision-making for restaurant owners.
6. Sentiment Data: Sentiment data on customer reviews, such as positive, negative, or neutral sentiment. This data can be used to identify specific aspects of the restaurant that customers like or dislike.

Overall, the data requirements for implementing the solution would depend on the specific implementation details and the scale of the project. A thorough analysis of the available data sources, data quality, and data integration requirements would be necessary to determine the optimal data approach.

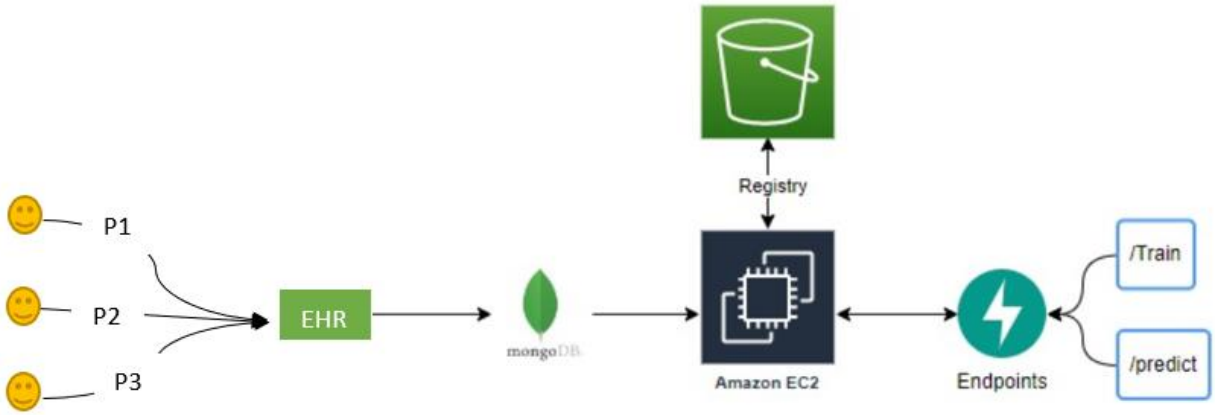
2.7 Tools used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn are used to build the whole model.

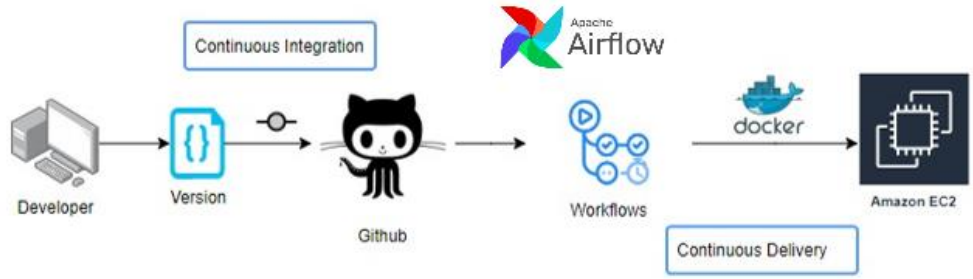
1. VS Code is used as IDE.
2. For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
3. AWS is used for deployment of the model.
4. MongoDB is used to retrieve, insert, delete, and update the database.
5. Front end development is done using HTML/CSS,
6. Flask API with modular coding is used for backend development.
7. GitHub is used as version control sys

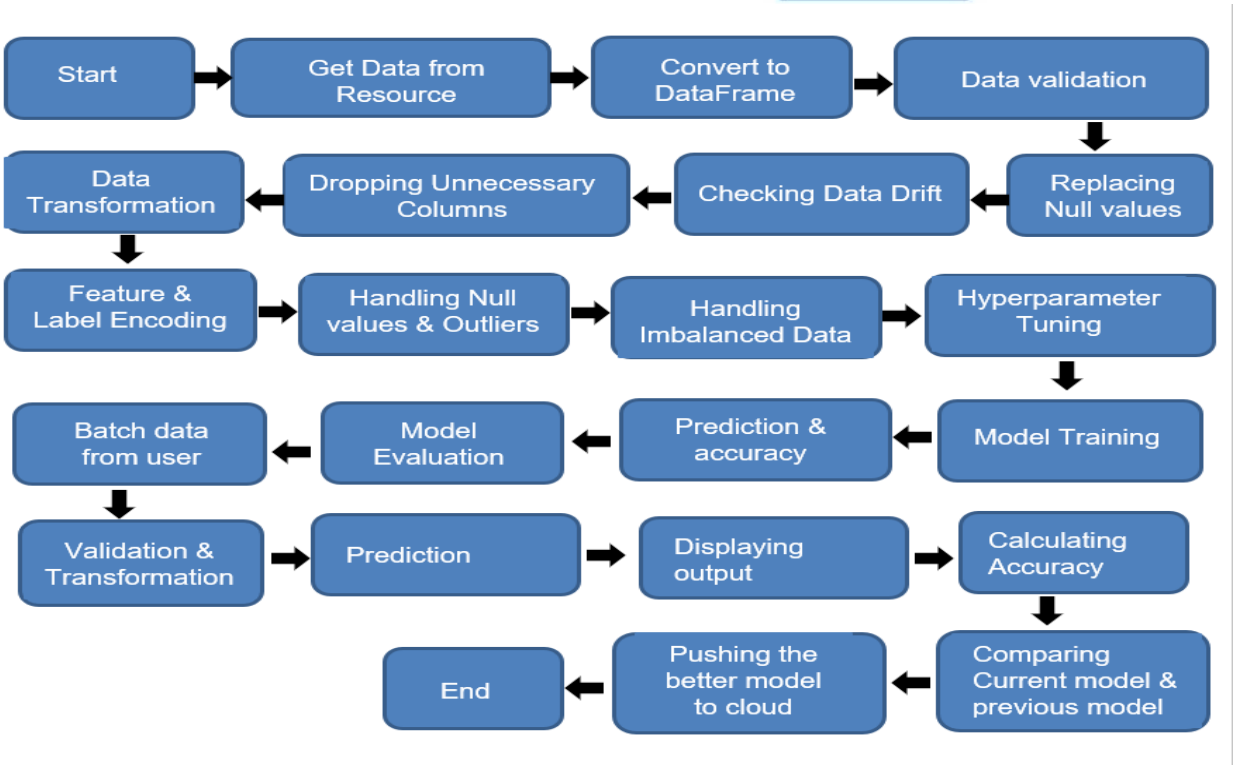
3. Design Details

3.1 Data Collections:

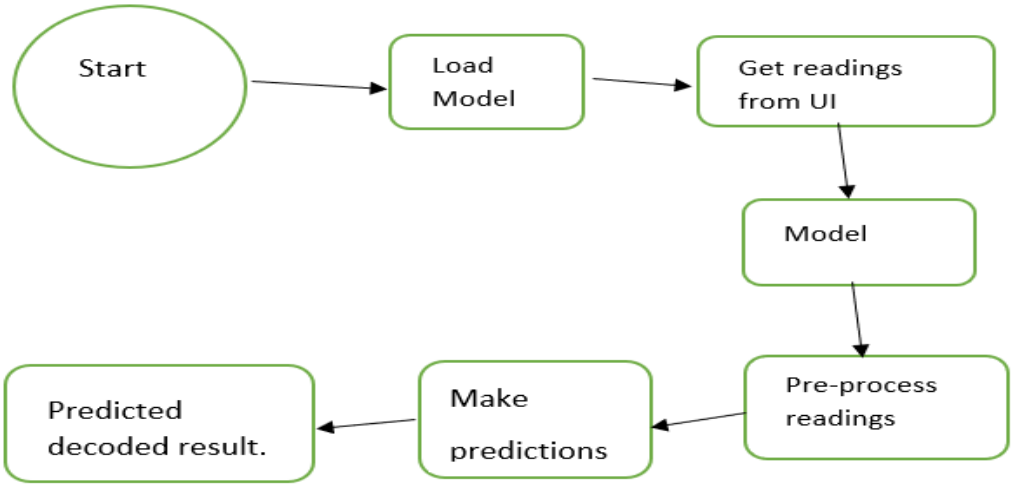


3.2 Project Architecture

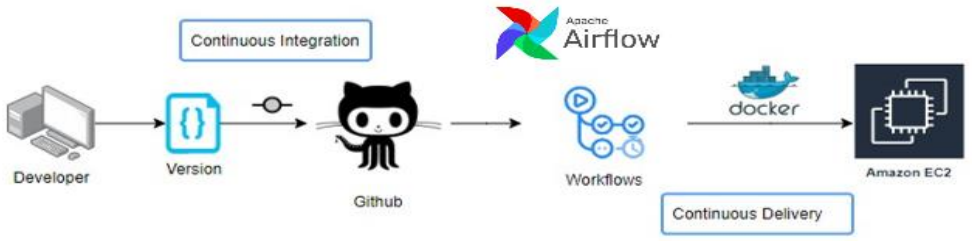


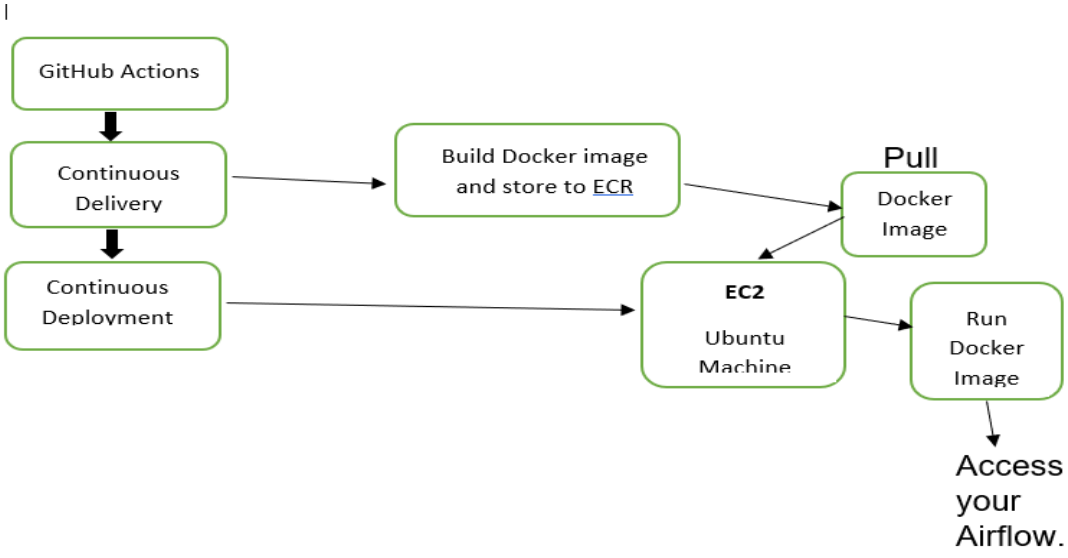


3.3 Real time prediction



3.4 Deployment Architecture





3.5 Event log

The system should log every event so that the user will know what process is running internally.

Initial Step-By-Step Description:

1. The System identifies at what step logging required
2. The System should be able to log each and every system flow.
3. Developer can choose logging method. You can choose database logging/ File logging as well.
4. System should not hang even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

3.6 Error Handling

Should errors be encountered, an explanation will be displayed as to what went wrong? An error will be defined as anything that falls outside the normal and intended usage

3.7 Performance

The performance of a machine learning model for restaurant rating prediction can be evaluated using various metrics. Some of the commonly used performance metrics are:

1. Mean Absolute Error (MAE): This metric measures the average absolute difference between the predicted and actual rating values. Lower values of MAE indicate better model performance.
2. Root Mean Square Error (RMSE): This metric measures the square root of the average squared difference between the predicted and actual rating values. Lower values of RMSE indicate better model performance.
3. R-squared (R2): This metric measures the proportion of variance in the dependent variable (rating) that can be explained by the independent variables (restaurant attributes). Higher values of R2 indicate better model performance.
4. Precision, Recall, and F1 Score: These metrics are used for evaluating the performance of classification models. Precision measures the proportion of true positive predictions among all positive predictions. Recall measures the proportion of true positive predictions among all actual positive instances. F1 score is the harmonic mean of precision and recall.
5. Accuracy: This metric measures the proportion of correct predictions among all predictions. Accuracy is a commonly used metric for evaluating classification models.

To improve the performance of the model, various techniques such as feature engineering, hyperparameter tuning, model selection, and data preprocessing can be used. Additionally, model ensembling, transfer learning, and deep learning algorithms can be used to further enhance the performance of the model.

Overall, the performance of the machine learning model for restaurant rating prediction would depend on various factors such as the quality and quantity of data, the model architecture and hyperparameters, and the evaluation metrics used. A thorough analysis of these factors and their impact on model performance would be necessary to achieve optimal performance.

3.7 Reusability

The code written and the components used should have the ability to be reused with noproblems.

3.8 Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

3.9 Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

3.10 Deployment

4 Conclusion

In conclusion, predicting restaurant ratings using machine learning algorithms can provide valuable insights for restaurant owners, customers, and other stakeholders in the food industry. The proposed solution involves using various restaurant attributes and customer feedback data to train a machine learning model that can predict the restaurant's rating. The solution can be implemented using various technical tools such as programming languages, cloud computing platforms, data storage systems, and user interfaces.

The success of the solution depends on the availability and quality of data, the accuracy of the machine learning algorithms, and the evaluation metrics used to measure the performance of the model. Additionally, the solution can be improved by incorporating more advanced techniques such as deep learning algorithms, transfer learning, and model ensembling.

Overall, implementing a machine learning-based solution for restaurant rating prediction can help restaurant owners improve their business operations and provide better services to their customers. It can also help customers make informed decisions when choosing a restaurant to dine at, and help other stakeholders in the food industry make data-driven decisions

5 References

1. https://www.kaggle.com/datasets/shrutimehta/zomato-restaurants-data
2. Scikit-Learn