

Restaurant Rating Prediction

-Tushar Jain




Introduction

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 of restaurants it has therefore become important to study the demography of a location. In the world of rising new technology and innovation, healthcare industry is advancing with the role of Artificial Intelligence. Machine learning algorithms can help to early detection of the disease and to improve the quality of the life. This study demonstrates the how different classification algorithms can forecasts the presence of the disease. Different classification algorithms such as Logistic regression, Random Forest, Decision Tree have been tested and compared to predict the better outcome of the model.

Objective

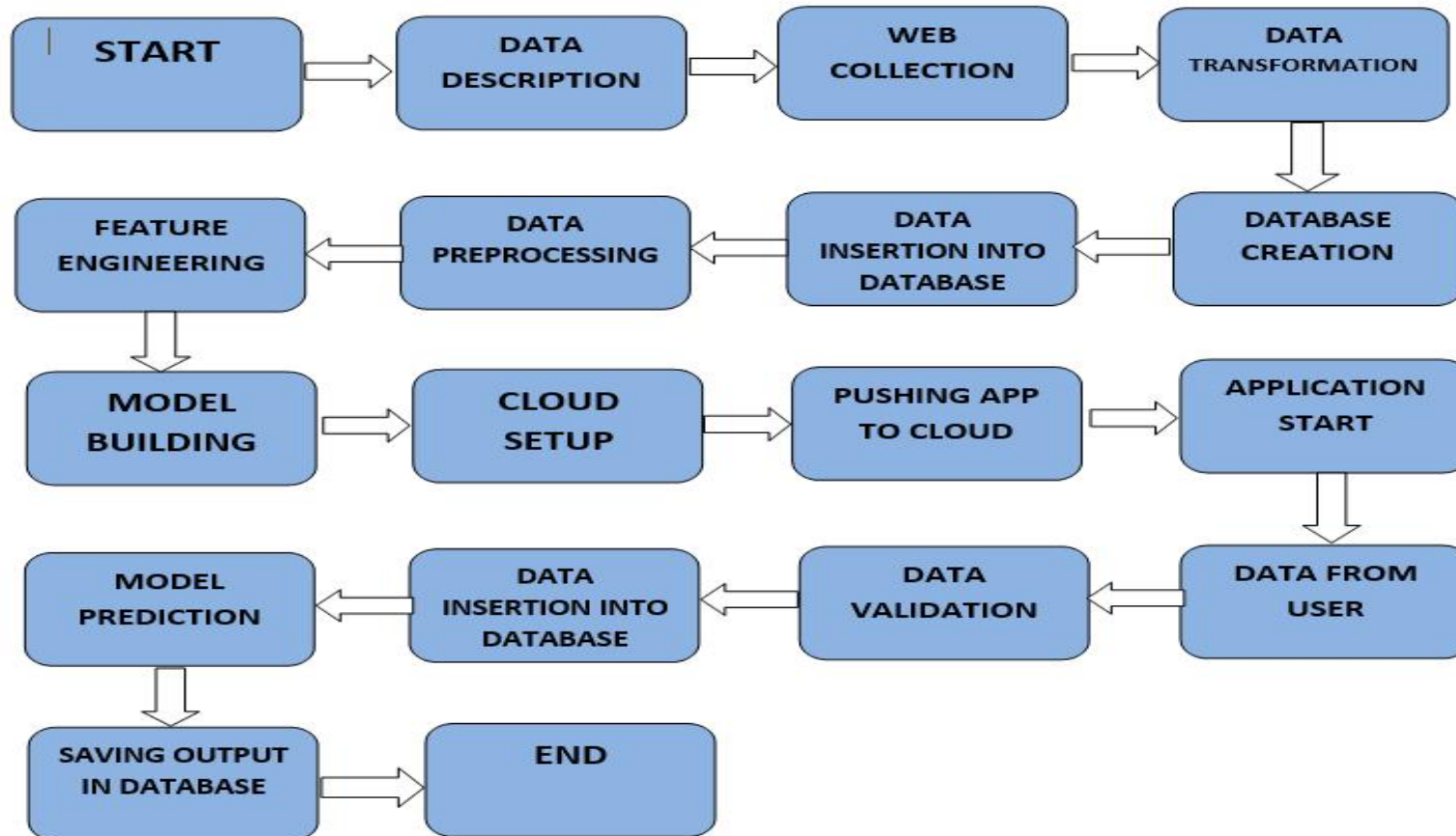
The Restaurant Rating Prediction is a machine learning based model which will help us to predict the rating of the restaurant in Bangalore. The dataset also contains reviews for each of the restaurant which will help in finding overall rating for the place.

The main goal of this project is to perform exploratory data analysis and later predict the rating of the restaurant.



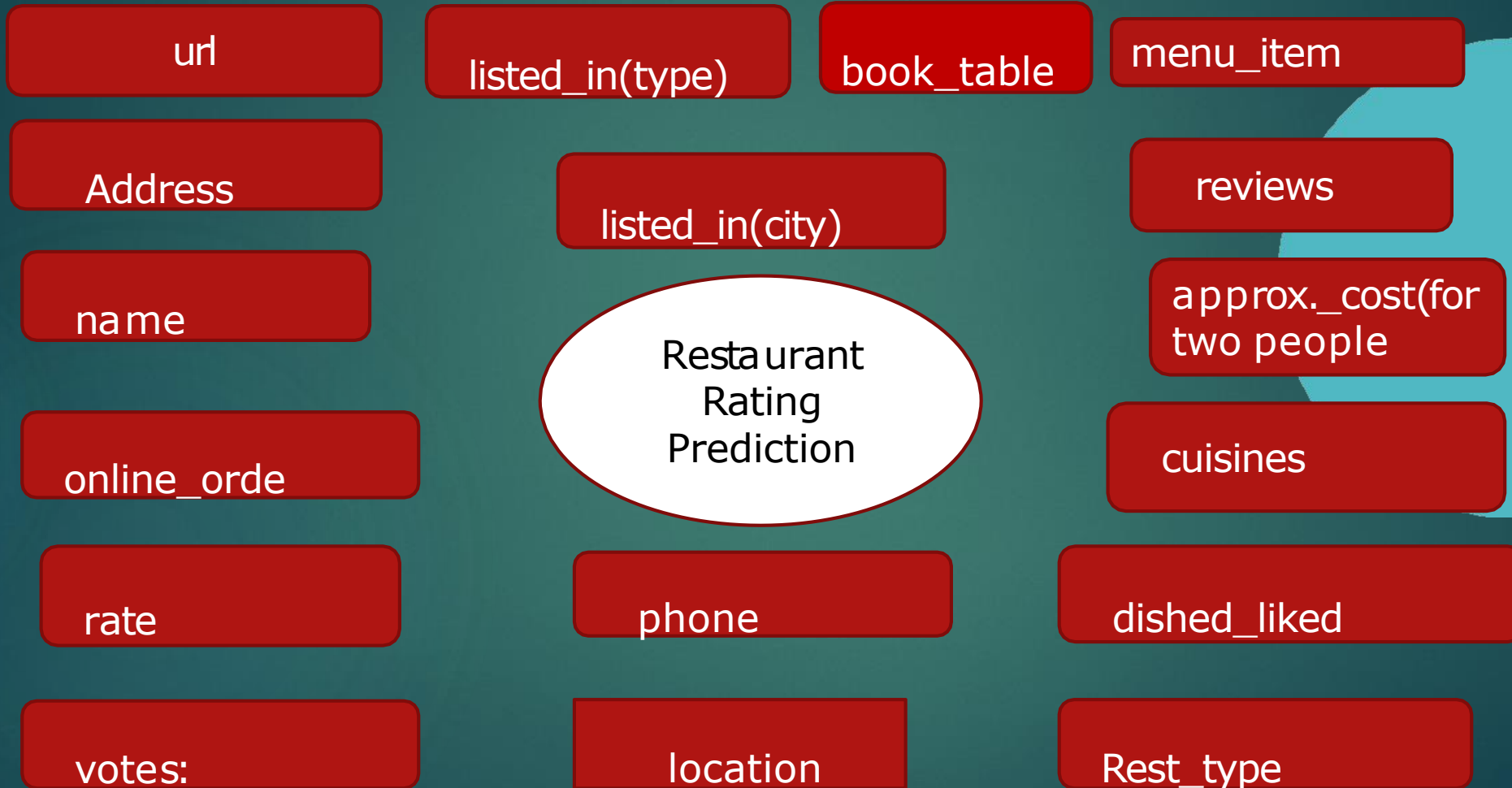
Architecture

4



Dataset

5



Data Analysis

6



DATA COLLECTION

In step 1, we collect data which is generally present in a database or on internet.



DATA PREPROCESSING

In step 2, we preprocess the data which involves data cleaning by handling outliers, null values etc.



EXPLORATORY DATA ANALYSIS

In step 3, we explore the data by performing univariate and bivariate analysis on the features.



FEATURE SELECTION

In step 4, we use feature selection techniques to filter out the most important features to perform model creation



MODEL CREATION AND EVALUATION

In step 5, we finally build models on our dataset and choose the model which gives the best accuracy.

Random Forest Model

7

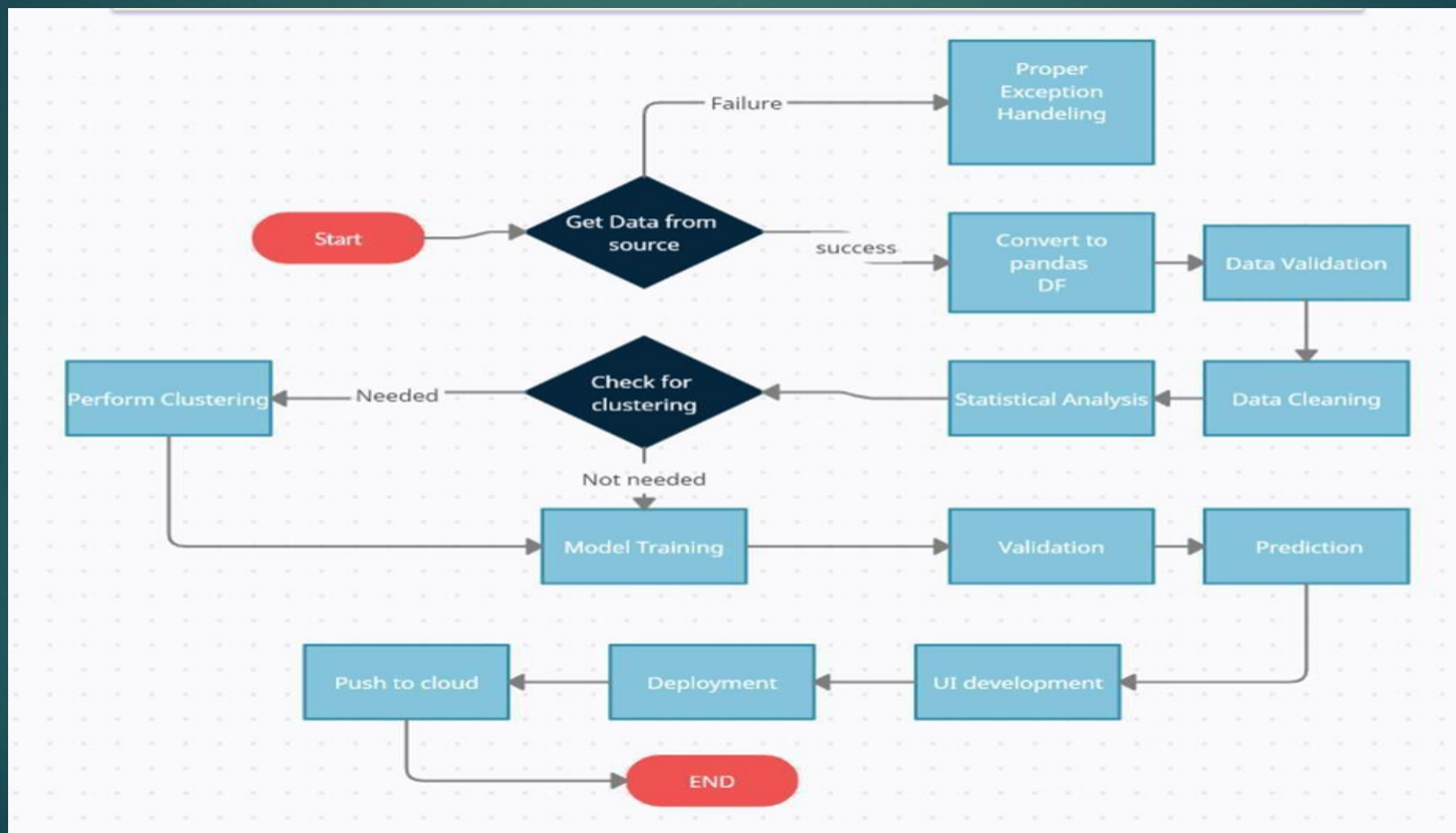
INTRODUCTION

- ▶ The random forest classifier is a supervised learning algorithm which we can use for regression and classification problems. It is among the most popular machine learning algorithms due to its high flexibility and ease of implementation.
- ▶ It is called Random Forest because it consists of multiple decision trees just as a forest has many trees. On top of that, it uses randomness to enhance its accuracy and combat overfitting, which can be a huge issue for such a sophisticated algorithm. These algorithms make decision trees based on a random selection of data samples and get predictions from every tree. After that, they select the best viable solution through votes.
- ▶ Random Forest Classifier being an ensemble algorithm tends to give more accurate results. This is because it works on the principle i.e. number of weak estimators when combined forms a strong estimator. Even if one or few decision trees are prone to noise, overall results would tend to be correct.

It gives us high accuracy as 87%.

MODEL TRAINING AND VALIDATION WORKFLOW

8



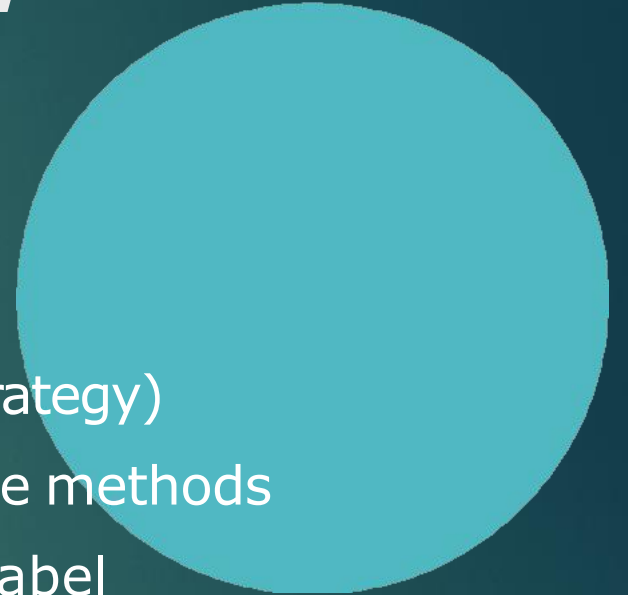
MODEL TRAINING AND VALIDATION WORKFLOW

Data Collection

- ▶ Zomato Restaurant Data Set from Kaggle.

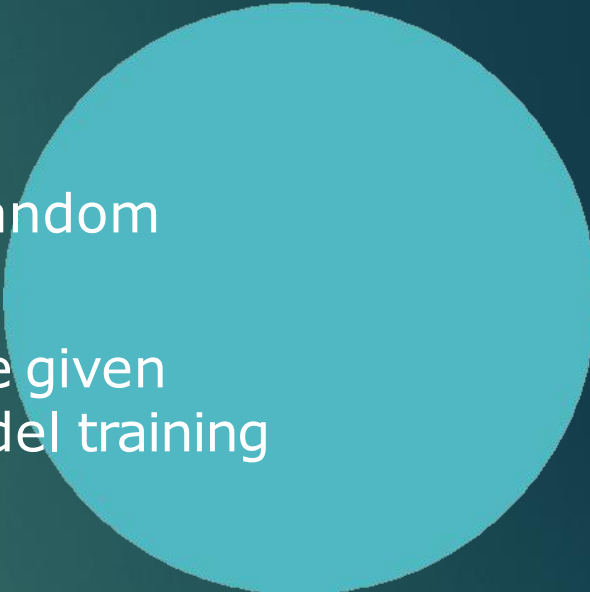
Data Pre-processing

- ▶ Missing values handling by Simple imputation (median strategy)
- ▶ Outliers' detection and removal by boxplot and percentile methods
- ▶ Categorical features handling by ordinal encoding and label encoding
- ▶ Feature scaling done by Standard Scalar method



MODEL TRAINING AND VALIDATION WORKFLOW

Model Creation and Evaluation

- ▶ Various classification algorithms like Logistic Regression, Random Forest, Decision Tree tested.
 - ▶ Random Forest, Decision Tree and Logistic regression were given better results. Random Forest was chosen for the final model training and testing.
 - ▶ Hyper parameter tuning was performed.
 - ▶ Model performance evaluated based on accuracy, confusion matrix, classification report.
- 

DEPLOYMENT

Model Deployment

- ▶ The final model is deployed using on Heroku using Flask framework



FAQs

12

Q1. What was type of data?

- ▶ It was a combination of numerical and categorical data.

Q2. How did you manage the null values in the dataset?

- ▶ Used `dropna()` attribute. Refer to code for better understanding.

Q3. What was the complete flow you followed in this project?

- ▶ Please refer to slide 4 for better understanding.

Q4. What were the techniques used for data pre-processing?

- ▶ Removing unwanted attributes.

- ▶ Dropping null values.
- ▶ Removing outliers.
- ▶ Visualizing relation between independent variables and dependent variables.
- ▶ Converting categorical data into numeric values.
- ▶ Scaling the data.

Q5. How did you train your model?

- ▶ Firstly, correlation was found among different variables.
- ▶ Then dataset was split into training and test size and different ML algorithms were used.
- ▶ Linear Regression, Decision Tree, Random Forest were used among which Random Forest gave the highest accuracy.



Q6. What was the accuracy of the best model observed?

- ▶ Random Forest showed the highest accuracy of 87% approx.

Q7. What challenges came up during the deployment of the model?

- ▶ An error in requirements.txt file can cause deployment failure.
- ▶ Requirement of correct and compatible versions of certain python libraries should be met.
- ▶ Cloning of Git repository should be done carefully. Use Git-LFS for files greater than 100mb.

Thank You

