RESTAURANT RECOMMENDATION SYSTEM

A UG PROJECT PHASE -1 REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Developed By

PADMALA SWATHI **19UK1A0593**

Under the esteemed guidance of

Dr. G. Ramesh

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

(Affiliated to JNTUH, Hyderabad)

Bollikunta, Warangal – 506005

2019- 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

BOLLIKUNTA, WARANGAL – 506005

2019- 2023

CERTIFICATE OF COMPLETION

UG PROJECT PHASE -1

This is to certify that UG Project phase-1 entitled "**RESTAURANT RECOMMENDATION SYSTEM**" is being developed *by* ***PADMALA SWATHI (19UK1A0593)*** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** to **Jawaharlal Nehru Technological University Hyderabad** during the academic year **2022-2023**, is a record of work carried out by her under the guidance and supervision.

Project Guide	**Head of the Department**
Dr. G. Ramesh	**Dr. R. Naveen Kumar**
(Assistant Professor)	(Professor)

External

ACKNOWLEDGEMENT

I wish to take this opportunity to express my sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO,** Principal, Vaagdevi Engineering College for making me available all the required assistance and for his support and inspiration to carry out this UG Project Phase-1 in the institute.

I extend my heartfelt thanks to **Dr.R.NAVEEN KUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing me necessary infrastructure and thereby giving me freedom to carry out the UG Project Phase-1.

I express heartfelt thanks to **Smart Bridge Educational Services Private Limited**, for their constant supervision as well as for providing necessary information regarding the UG project Phase-1 and for their support in completing the UG project Phase-1.

I express heartfelt thanks to the guide **Dr.G. Ramesh,** Assistant professor, Department of CSE for his constant support and giving necessary guidance for completion of this UG project Phase-1.

Finally, I express my sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

PADMALA SWATHI **(19UK1A0593)**

ABSTRACT

The Recommendation system is the unavoidable thing for whatever we buy or go to the new place. Restaurants also need recommendation systems in terms of attracting more customers in the management side and tasting favorite, famous food in the restaurant in customers side. In reality finding the favorite food and famous food especially in new area is a challenging task. In this project, we present the recommendation system for restaurants based on ratings. With addition to that we build the popularity-based recommender model for recommending restaurants to the customers. The output of the model may be recommending most popular restaurants to the customers. The aim is to create a content-based recommender system in which when we will write a restaurant name, the Recommender system will look at the reviews of other restaurants, and the System will recommend us other restaurants with similar reviews and sort them from the highest-rated. The main people who are going to benefit from this recommendation system are the tourists, who are new to a city. Most of the tourists always love to visit famous restaurants in a particular city during their visit. Otherwise, it can be heavily used by people belonging to the same city, to see if any new restaurant is recommended based on their activity.

Key words: Recommendation, Customers, Restaurants, Content-based Recommendation.

TABLE OF CONTENTS

LIST OF FIGURES

1.INTRODUCTION

1.1 Overview

This project is a restaurant recommendation system (RRS). A Recommendation System is an information filtering system that seeks to predict the rating a user would give for the item (in this case a restaurant). RRS is an on-line system to search restaurants. Visitors can browse all restaurants in L.A, and. Get information about restaurant name, type, rating, price. The functions include searching restaurants, viewing recommendations. Recommendation systems are important for increasing business revenue and giving users the ability to find desired restaurants of their taste. The system is challenging because many users don't give ratings and we have new restaurants and users added to the system every day. In order to improve restaurant rating system, we need to predict the rating for the restaurant which are not rated. So it is important to build recommendation system for sparse rated restaurants. For this recommendation model, all that users have to input is a restaurant name that they have previously enjoyed visiting into the model and it will generate a list of the 10 most recommended restaurants based on the highest cosine similarity scores to that particular restaurant. For the content-based recommendation model**,** it works by recommending restaurants to users based on similar restaurant categories and dominant topic keywords, thus suggesting restaurants that align with a user's preferences.

1.2 Purpose

The purpose of this system is to let people get ideas about which restaurant will be great for them. This system can give people some suggestions; also you can get others' opinions from this site. Further more, you can find the best restaurants by viewing the ratings page, which gathers many members' experience and response. This system is designed for people to search the information you send, and response all those restaurants matched the customers' request. Except viewing other's opinions, you can give suggestions by rating restaurants to other people. This system is like a communication bulletin for people who love to eat. In this site, the customer need to search restaurants by their names. They will get a page describing the related names of the restaurants and their type and ratings.

2. LITERATURE SURVEY

2.1 Existing problem and Existing approaches or methods

As we are users of recommendation applications, people care more about how we will like a restaurant. It is very common that we hang out with families, friends, and co-workers. when comes to lunch or dinner time. In the past, people obtained suggestions for restaurants from friends. Although this method is straightforward and user-friendly, it has some severe limitations. First, the recommendations from friends or other common people are limited to those places they have visited before. Thus, the user is not able to gain information about places less visited by their friends. Besides that, there is a chance of users not liking the place recommended by their friends.

2.2 Disadvantages of existing system

- Collecting large amount of data set.
- Large number of training data and annotations are needed which may not be practical in some problems.

2.3 Proposed solution

Here we are creating a content-based recommendation system. The aim is to create a content-based recommender system in which when we will write a restaurant name, the Recommender system will look at the reviews of other restaurants, and the System will recommend us other restaurants with similar reviews and sort them from the highest-rated. The main people who are going to benefit from this recommendation system are the tourists, who are new to a city. Most of the tourists always love to visit famous restaurants in a particular city during their visit. Otherwise, it can be heavily used by people belonging to the same city, to see if any new restaurant is recommended based on their activity.

2.4 Proposed algorithams

I. k-nearest neighbor algorithm:

- It is a supervised machine learning algorithm. The algorithm can be used to solve both classification and regression problem statements. The number of nearest neighbor's to a new unknown variable that has to be predicted or classified is denoted by the symbol 'K'.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- K-Nearest Neighbors (KNN) is one of the simplest algorithms used in Machine Learning for regression and classification problem. KNN algorithms use data and classify new data points based on similarity measures (e.g. distance function). The data is assigned to the class which has the nearest neighbors.
- It's also worth noting that the KNN algorithm is also part of a family of ―lazy learning‖ models, meaning that it only stores a training dataset versus undergoing a training stage. This also means that all the computation occurs when a classification or prediction is being made. Since it heavily relies on memory to store all its training data, it is also referred to as an instance-based or memory-based learning method.

- The K-NN working can be explained on the basis of the below algorithm:

Step-1: Select the number K of the neighbors

Step-2: Calculate the Euclidean distance of K number of neighbors

Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

Step-6: Our model is ready.

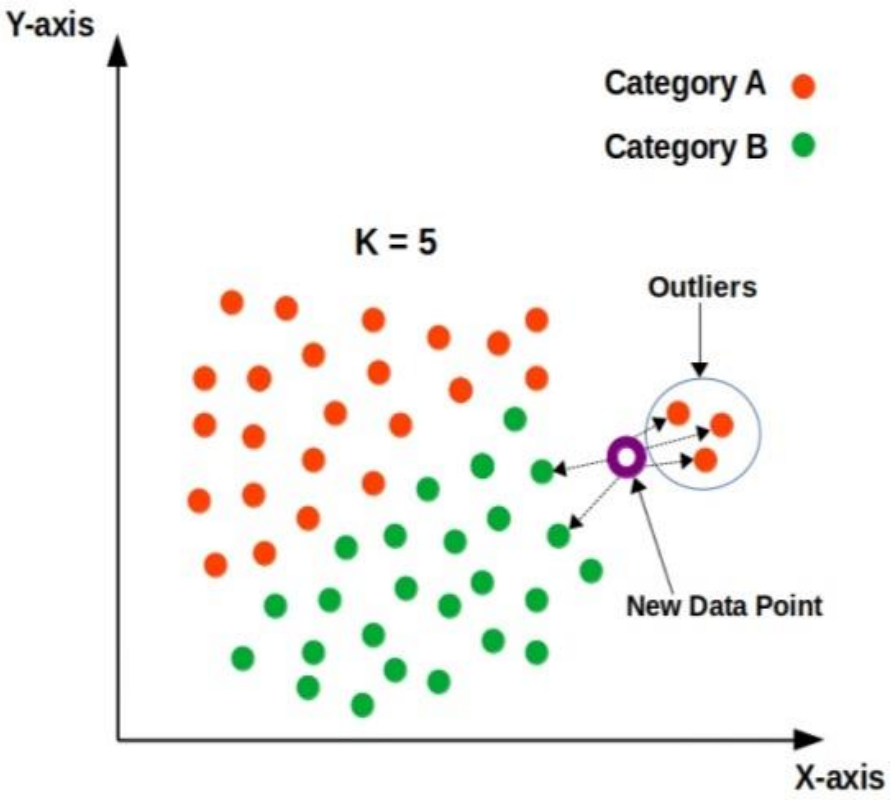
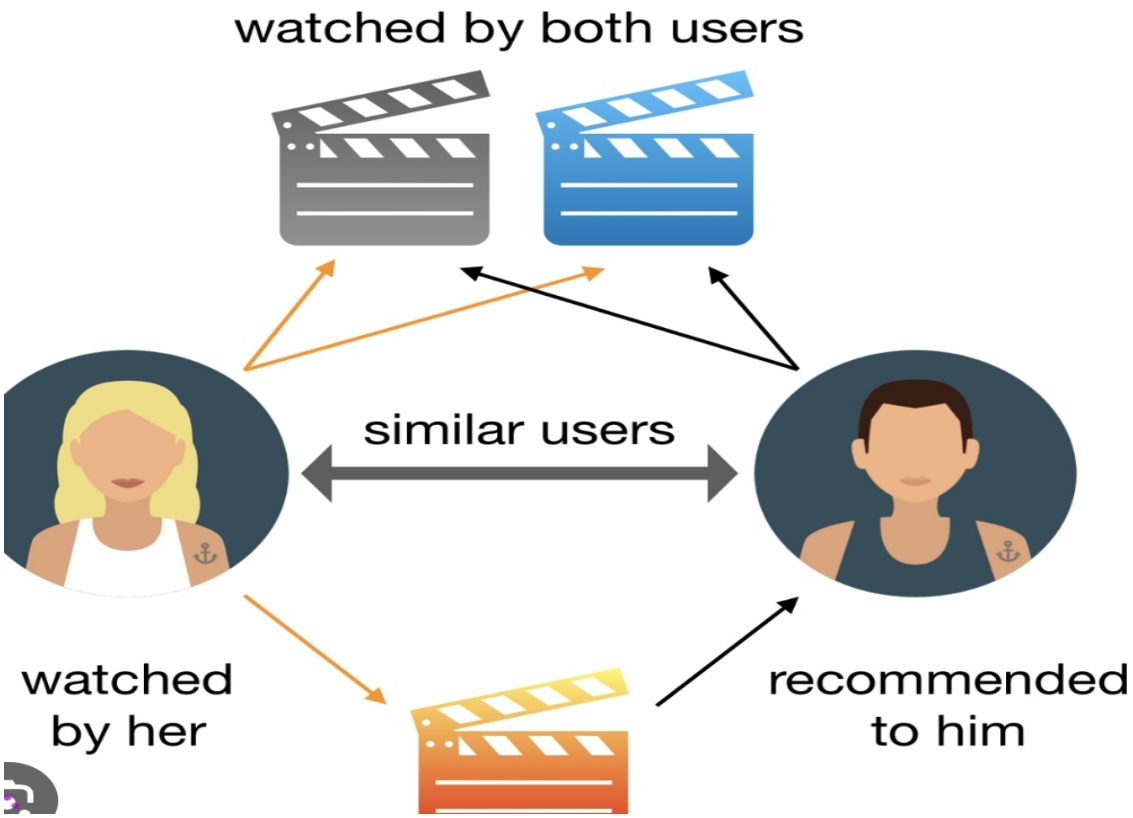


Figure 1:KNN Algorithm

II.Linear Regression

The most common is the linear regression algorithm. The linear regression algorithm is used to find the best linear approximation to a data set. In a recommender system, this algorithm is used to predict how a user will rate an item based on their past ratings



Linear Regression is a supervised machine learning algorithm where the predicted output

Figure 2: Linear Regression

3.THEORITICAL ANALYSIS

3.1 Block Diagram

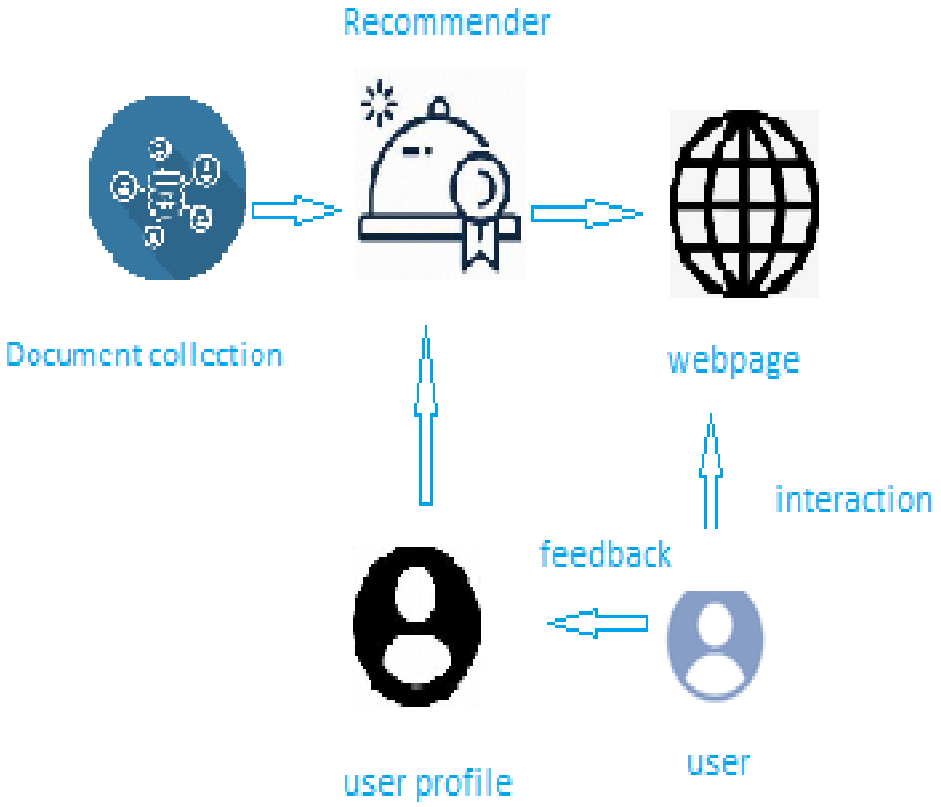


Figure3. Block diagram

3.2 Hardware and Software

Software Requirements:

To complete this project, you must require the following software's, concepts, and packages

Anaconda navigator

Python packages:

- pandas
- matplotlib
- seaborn
- plotly
- numpy
- scikit-image
- scikit-learn
- Flask

Hardware Requirements

- Processor : Intel Core i3
- Hard Disk Space : Min 100 GB
- Ram : 4 GB
- Display : 14.1 "Color Monitor(LCD, CRT or LED)
- Clock Speed : 1.67 GHz

4.EXPERIMENTAL INVESTIGATION

For developing the project the team has completed several tasks:

1.Data Collection.

ML depends heavily on data, without data, a machine can't learn. It is the most crucial aspect that makes algorithm training possible. In Machine Learning projects, we need a training data set. It is the actual data set used to train the model for performing various actions.

You can collect datasets from different open sources like kaggle.com, data.gov; UCI machine learning repository etc. The dataset used for this project was obtained from Kaggle.

2. Data Pre- processing.

Data Pre-processing includes the following main tasks

- Import the Libraries.
- Importing the dataset.
- Analysis the data
- Taking care of missing Data
- Exploratory Data Analysis
- Data Visualization.
- Splitting Data into Train and Test

3.Model Building

The model building process involves setting up ways of collecting data, understanding and paying attention to what is important in the data to answer the questions you are asking, finding a statistical, mathematical or a simulation model to gain understanding and make predictions. Model Building Includes:

- Import the model building libraries
- Initialising the model
- Training the model
- Model evalution
- Save the model

4. Application Building

1. Create an HTML file
2. Build a Python Code
3. Run the app in local browser
4. Show casting the prediction on UI

5.FLOW CHAT

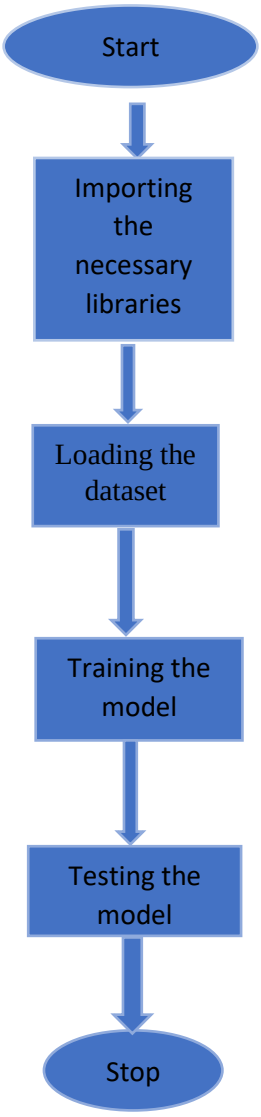
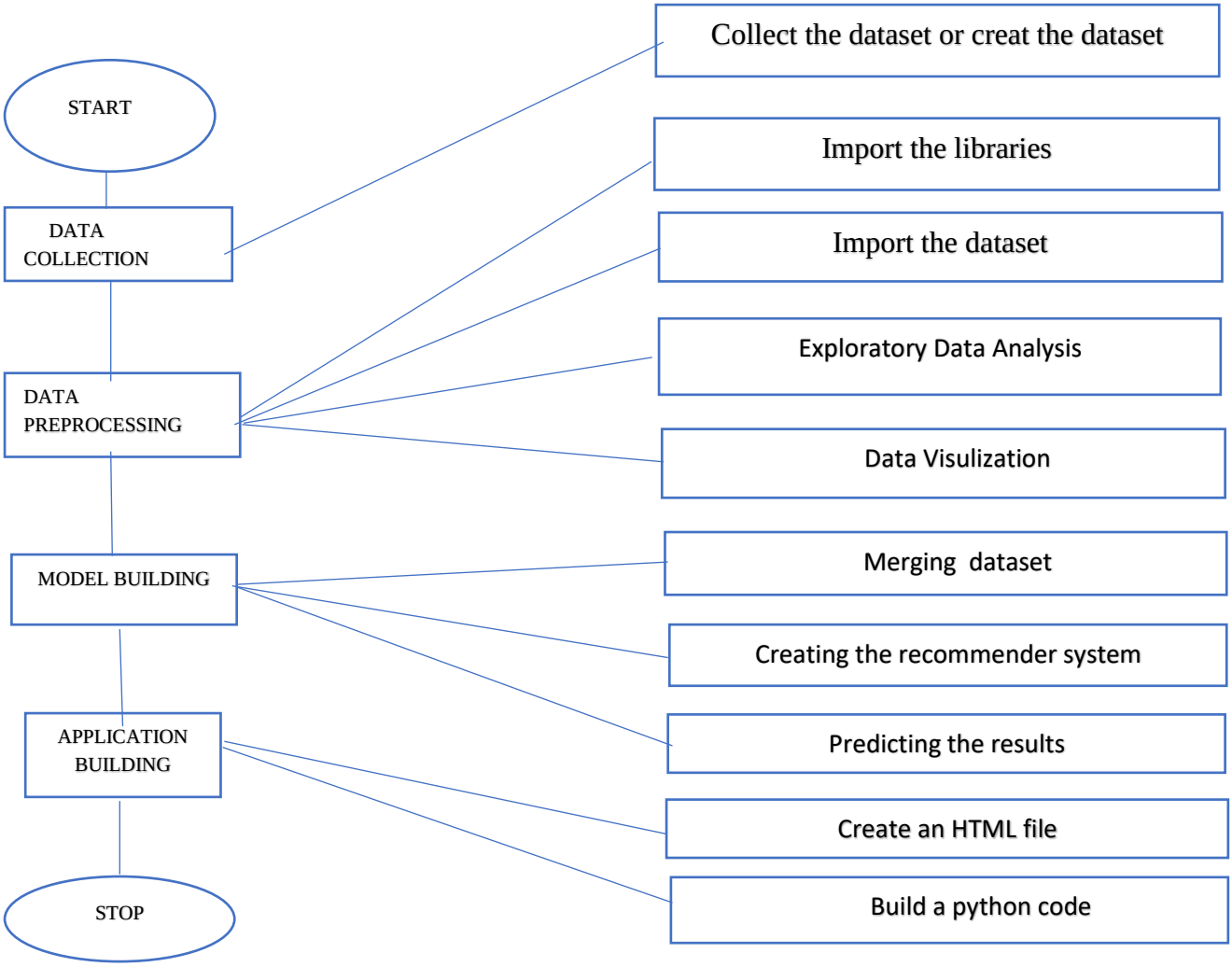


Figure.4 Flowchart

6.DESIGN



Pre requisties

To complete the project successfully, you need to install following software & packages**:**

Install Anaconda IDE/Anaconda Navigator

In order develop a solution to this problem statement, we need an environment to write and test the code**.**

We use Anaconda IDE (Integrated Developing Environment).

Refer to the below link to download & install Anaconda Navigator.

To Build Machine Learning Models You Must Require The Following Packages

Numpy:

It is an open-source numerical Python library. It contains a multidimensional array and matrix data structures and can be used to perform mathematical operations

Matplotlib and Seaborn

Matplotlib is mainly deployed for basic plotting. Visualization using Matplotlib generally consists of bars, pies, lines, scatter plots and so on. Seaborn: Seaborn, on the other hand, provides a variety of visualization patterns. It uses fewer syntax and has easily interesting default themes.

Flask:

Web framework used for building Web applications

If you are using anaconda navigator, follow below steps to download required packages:

Open anaconda prompt.

Type “pip install pandas” and click enter.

Type “pip install matplotlib” and click enter.

Type “pip install seaborn” and click enter.

Type “pip install plotly” and click enter.

Type “pip install numpy” and click enter.

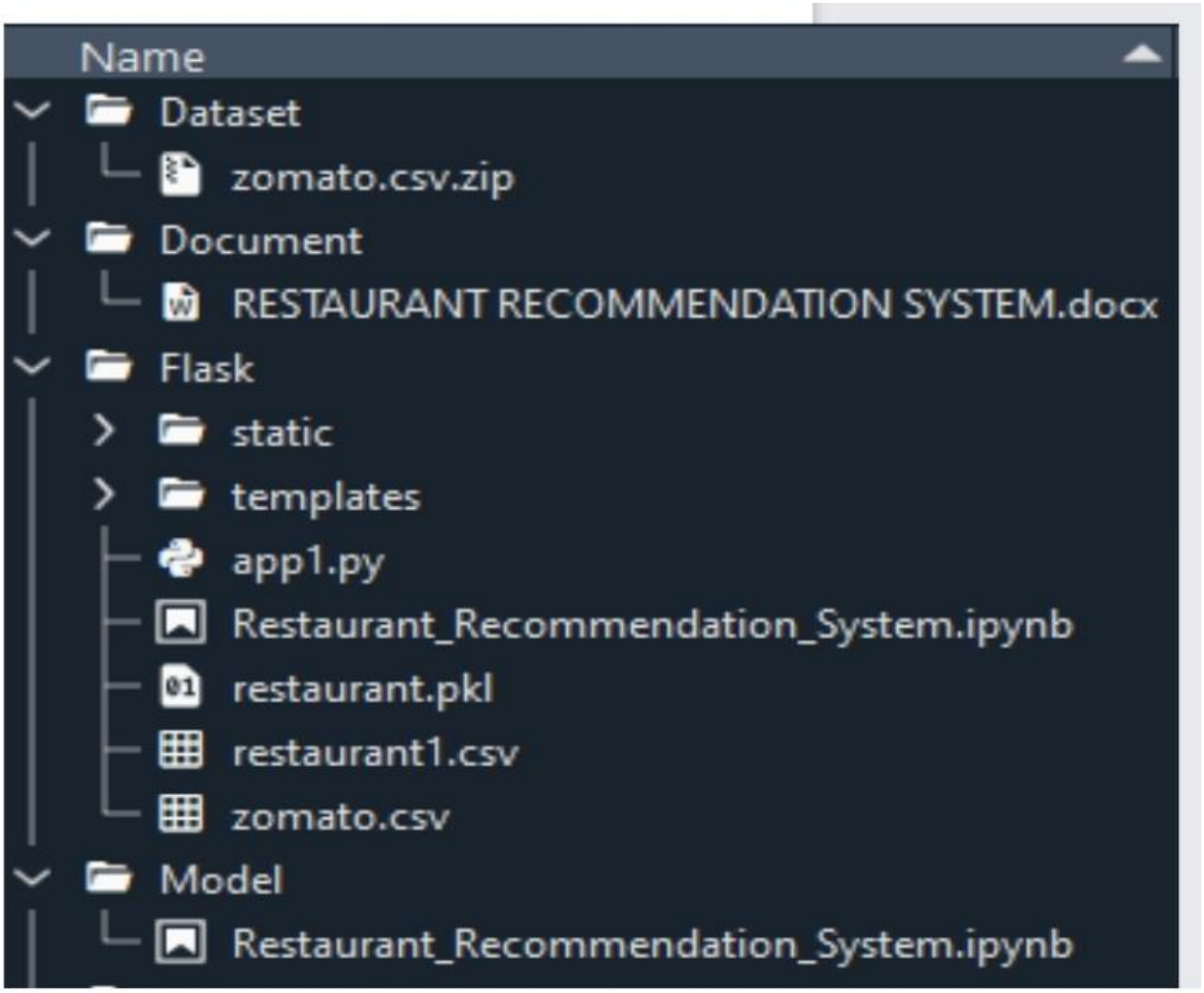
Type “pip install scikit-image” and click enter.

Type “pip install scikit-learn” and click enter.

Type “pip install Flask” and click enter.

7.PROJECT STRUCTURE

- User interacts with the UI (User Interface) to enter the input features.
- Entered features/input is analysed by the model which is integrated
- Once model analyses the entered inputs, the prediction is showcased on the UI.



- **Figure 5**: Project Structure

.

8.CONCLUSION

The main objective of the study is to develop the restaurant recommendation system using machine learning with the web interface that can act as a application for the customers. This application is used for the users to predict the suitable restaurant and find out which dish is famous in region wise and in person. This application ensures the availability of ratings to the customers. The popularity based and collaborative based filtering makes the recommendation more efficient so that each user can use this application for their easy prediction of restaurant. Most the case user need the restaurant with their nearby location. We also solving that issue by adding the restaurant location in our dataset. So that our machine learning algorithm easily predicts the restaurant for the customer with their present location. This restaurant recommendation system web application will provide user a better experience in searching of restaurant with short amount of time and nearby location. This will decrease the user’s effort and makes the time more precious.

9.FUTURE SCOPE

To build more friendly graphical interfaces, the next goal for a further project is to improve the performance of system and improve the member's benefit such as providing more functions for member, like online reservation function, online order menu, and website, etc.

RESTAURANT RECOMMENDATION SYSTEM

A UG PROJECT PHASE -2 REPORT

Submitted to

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

In partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Developed By

PADMALA. SWATHI **19UK1A0593**

Under the esteemed guidance of

Dr. G. Ramesh

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

(Affiliated to JNTUH, Hyderabad)

Bollikunta, Warangal – 506005

2019- 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VAAGDEVI ENGINEERING COLLEGE

BOLLIKUNTA, WARANGAL – 506005

2019- 2023

CERTIFICATE OF COMPLETION

UG PROJECT PHASE -2

This is to certify that UG Project phase-2 entitled "**RESTAURANT RECOMMENDATION SYSTEM**" is being developed by ***PADMALA SWATHI (19UK1A0593)*** in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering to Jawaharlal Nehru Technological University Hyderabad during the academic year 2022-2023, is a record of work carried out by her under the guidance and supervision.

Project Guide	**Head of the Department**
Dr. G. Ramesh	**Dr. R. Naveen Kumar**
(Assistant Professor)	**(Professor)**

External

ACKNOWLEDGEMENT

I wish to take this opportunity to express my sincere gratitude and deep sense of respect to our beloved **Dr.P.PRASAD RAO,** Principal, Vaagdevi Engineering College for making me available all the required assistance and for his support and inspiration to carry out this UG Project Phase-2 in the institute.

I extend my heartfelt thanks to **Dr.R.NAVEEN KUMAR**, Head of the Department of CSE, Vaagdevi Engineering College for providing me necessary infrastructure and thereby giving me freedom to carry out the UG Project Phase-2.

I express heartfelt thanks to **Smart Bridge Educational Services Private Limited**, for their constant supervision as well as for providing necessary information regarding the UG project Phase-2 and for their support in completing the UG project Phase-2.

I express heartfelt thanks to the guide **Dr.G. Ramesh,** Assistant professor, Department of CSE for his constant support and giving necessary guidance for completion of this UG project Phase-2.

Finally, I express our sincere thanks and gratitude to my family members, friends for their encouragement and outpouring their knowledge and experience throughout the thesis.

PADMALA. SWATHI (19UK1A0593)

TABLE OF CONTENTS

LIST OF FIGURES

1.INTRODUCTION

A recommendation system or recommender system is a type of information filtering system that uses various features given about user and product and tries to predict the most similar pairs to identify the best products according to user taste that a user is most likely to consume and return positive feedback.

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. Here we are building a model by applying various machine learning algorithms find the best accurate model. So we can easily find which restaurant is best in the new places to vigistors. On our Dataset , we have applied Random Forest Regression and KNN algorithms.

The abbreviation KNN stands for ―K-Nearest Neighbour‖. It is a supervised machine learning algorithm. The algorithm can be used to solve both classification and regression problem statements. The number of nearest neighbours to a new unknown variable that has to be predicted or classified is denoted by the symbol 'K'.

UG Project Phase-2 involves all the coding and implementation of the design which we have retrieved from UG Project Phase-1. All the implementation is done and conclusions are retrieved in this phase. We will also work on the applications, advantages, and disadvantages of the project in this phase. Future scope of the project will be also discussed in the UG Project Phase-2

2.CODE SNIPPETS

2.1 .MODEL CODE:

Data Preprocessing

Data Pre-processing includes the following main tasks

1.Import the Libraries.

2.Importing the dataset.

3.Checking for Null Values.

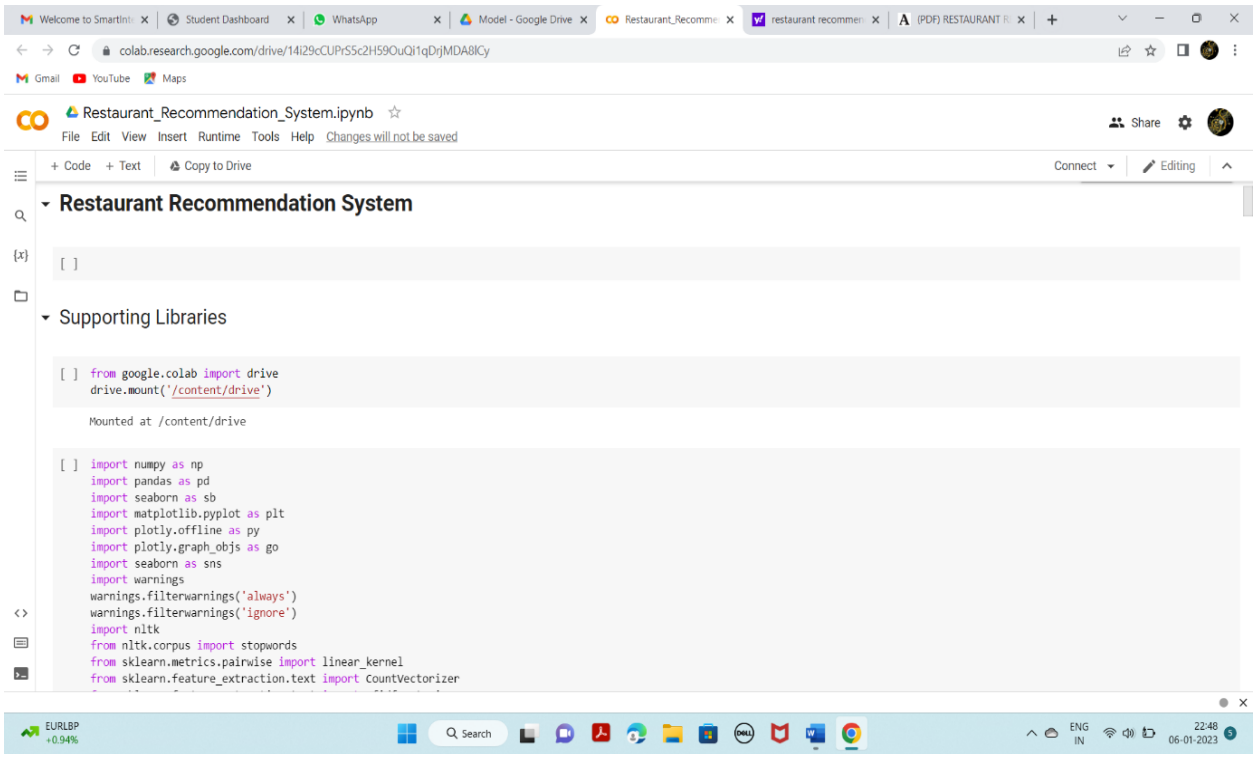
4.Data Visualization.

5.Label Encoding.

6.OneHot Encoding.

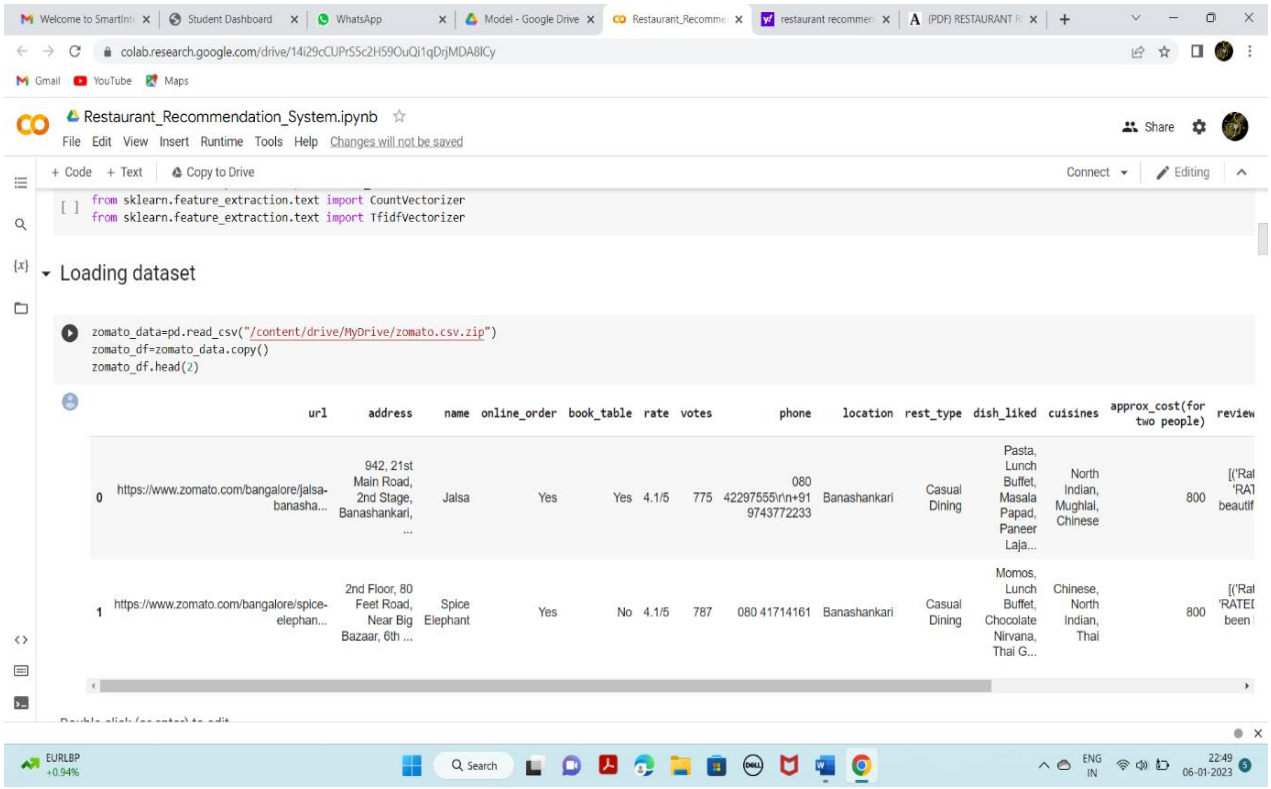
7.Splitting Data Into Train and Test.

Import the below essential libraries for data pre-processing and creating recommendation system. Pandas and NumPy are used for data pre-processing and cleaning. Seaborn, Plotly and Matplotlib helped in creating visual graphics and bar plots for the dataset. Also, since there would be cleaning of text data (reviews) as well, therefore for that we will use nltk and sklearn library.



Our dataset format might be in .csv, excel files, .txt, json, etc. We can read the dataset with the help of pandas.

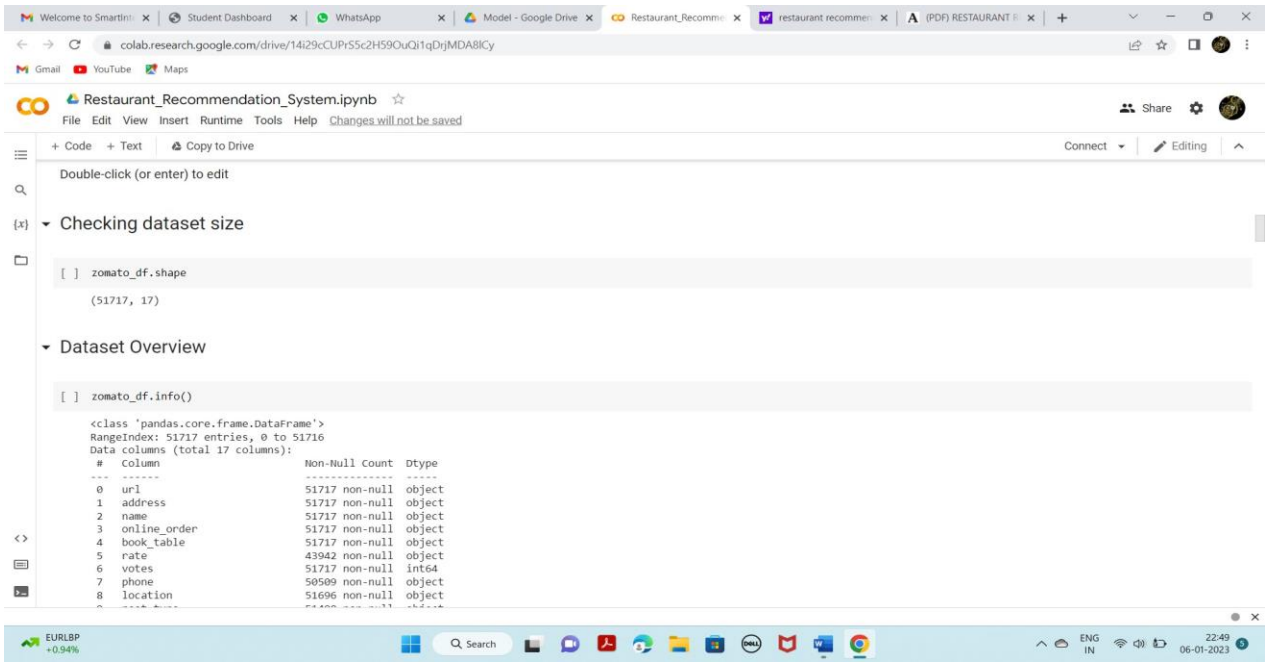
In pandas we have a function called read_csv () to read the dataset. As a parameter we have to give the directory of csv file.



Understanding Overview of features

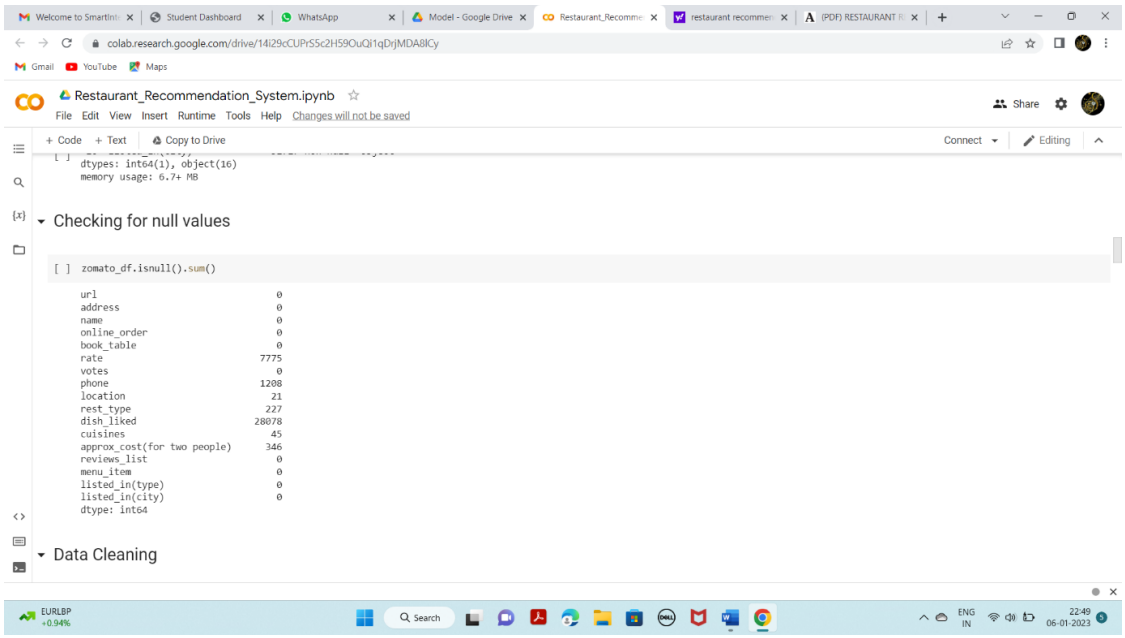
- How the information is stored in a DataFrame or Python object affects what we can do with it and the outputs of calculations as well. There are two main types of data those are numeric and text data types.
- Numeric data types include integers and floats.
- Text data type is known as Strings in Python, or Objects in Pandas. Strings can contain numbers and / or characters.
- For example, a string might be a word, a sentence, or several sentences.
- Will see how our dataset is, by using info () method.

.

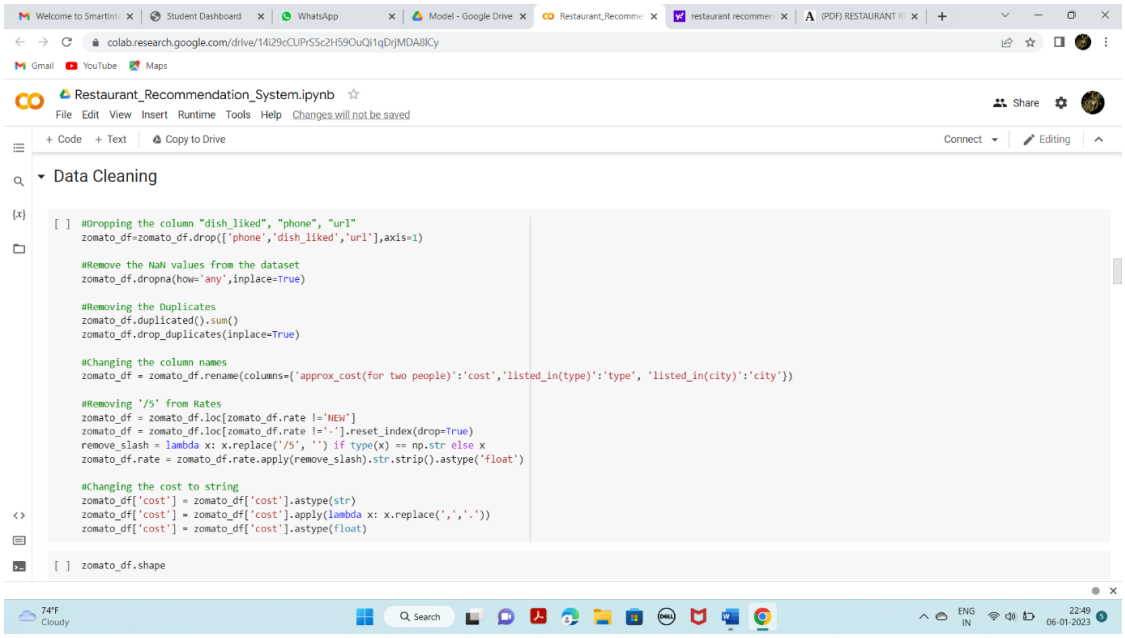


As you can see in our dataset, except ‘votes’, all other features are categorical data, but it is not necessary that all the continuous data which we are seeing has to be continuous in nature. There may be a case that some categorical data is in the form of numbers but when we perform info () operation we will get numerical output. So, we need to take care of those type of data also

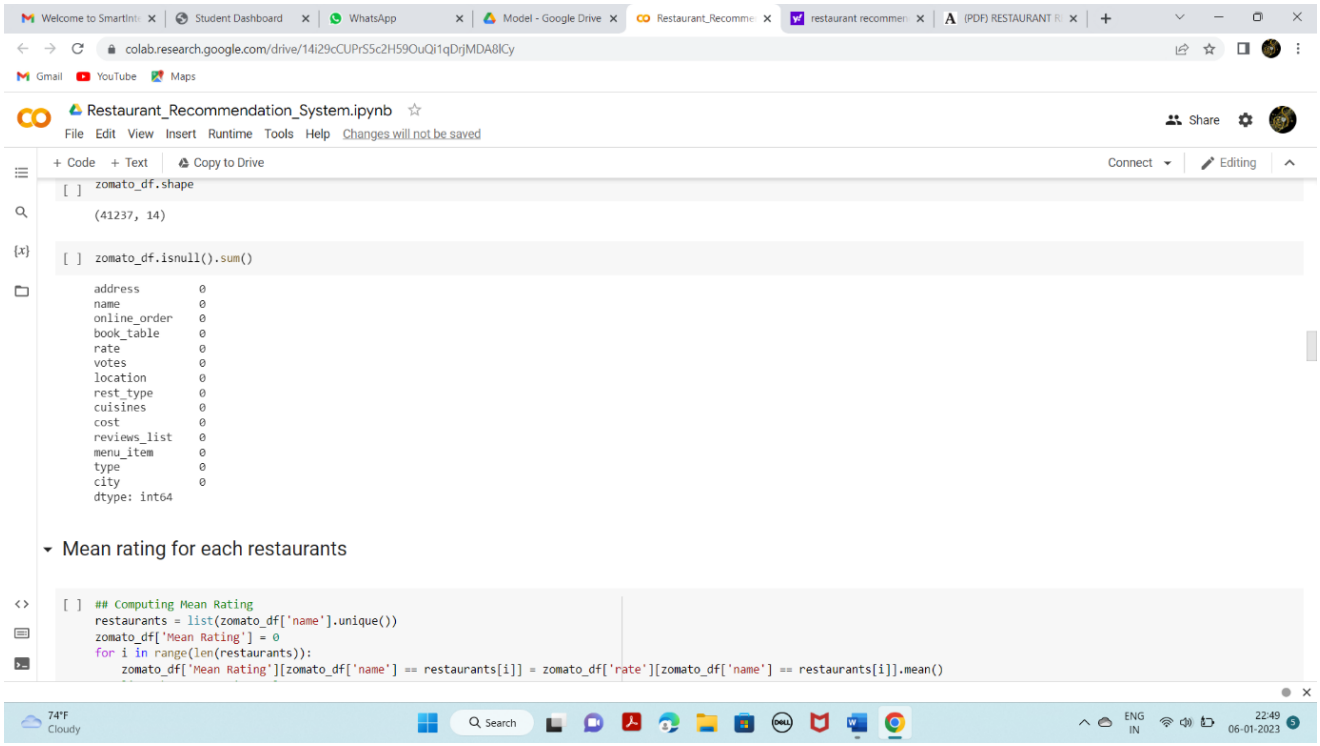
Checking for null values in the dataset



Data cleaning as our dataset contains null values and some special characters



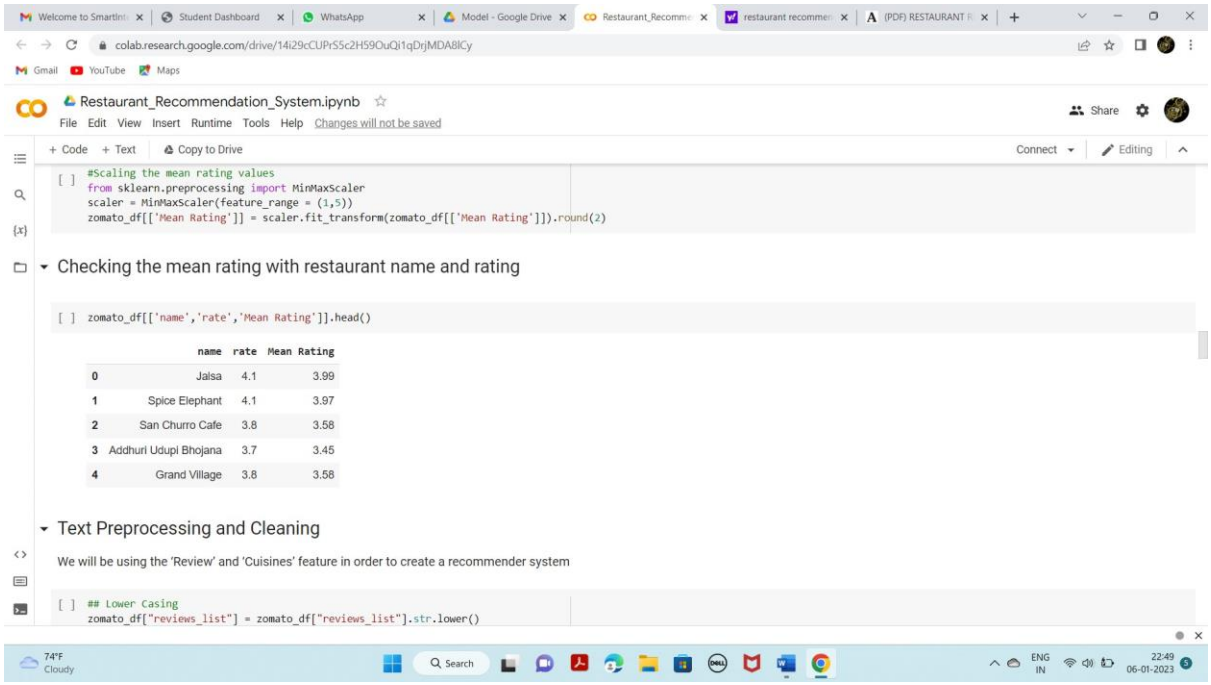
Checking for null values after cleaning & data Processing

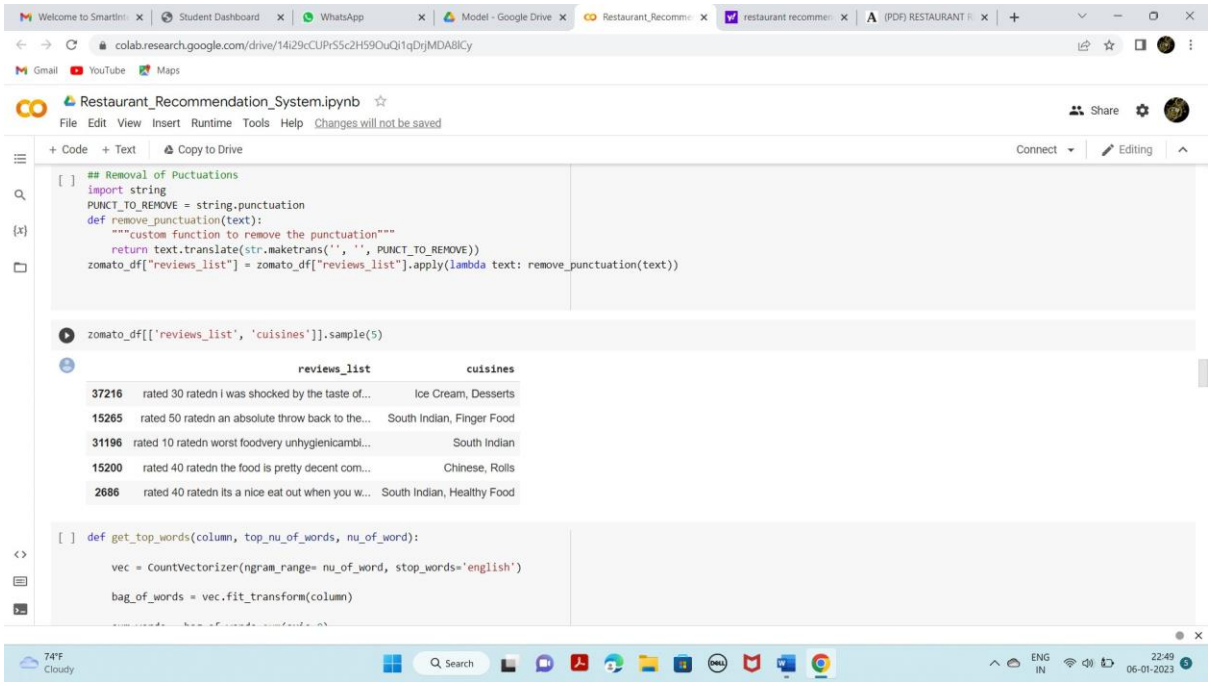


Checking mean rating with restaurant name and rating for each restaurant using below line codes

We will be using the ‘Review’ and ‘Cuisines’ feature in order to create a recommender system. So we need to prepare and clean the text in those columns.

Operations performed: Lower Casing, Removal of Punctuations, Removal of Stop words, Removal of URLs, Spelling correction





We will be using the ‘Review’ and ‘Cuisines’ feature in order to create a recommender system. So we need to prepare and clean the text in those columns.

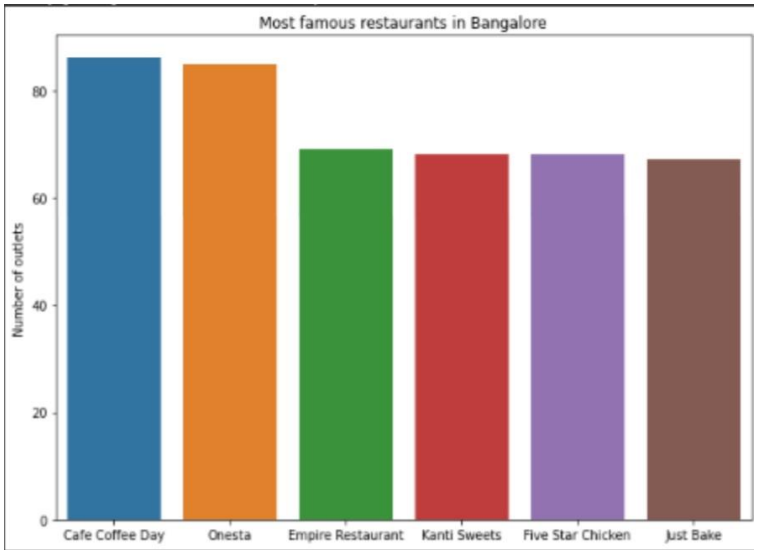
Operations performed: Lower Casing, Removal of Punctuations, Removal of Stop words, Removal of URLs, Spelling correction

To visualize the dataset, we need libraries called Matplotlib, Seaborn. The Matplotlib library is a Python 2D plotting library which allows you to generate plots, scatter plots, histograms,

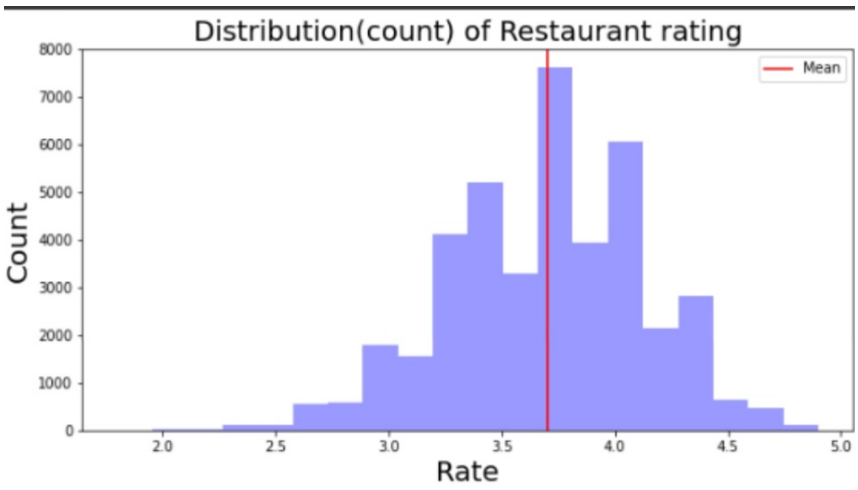
bar charts etc.

Let’s visualize our data using Matplotlib and seaborn library.

At first, we will be plotting a bar plot using matplotlib for showing the top 6 restaurants in Bangalore by value counts

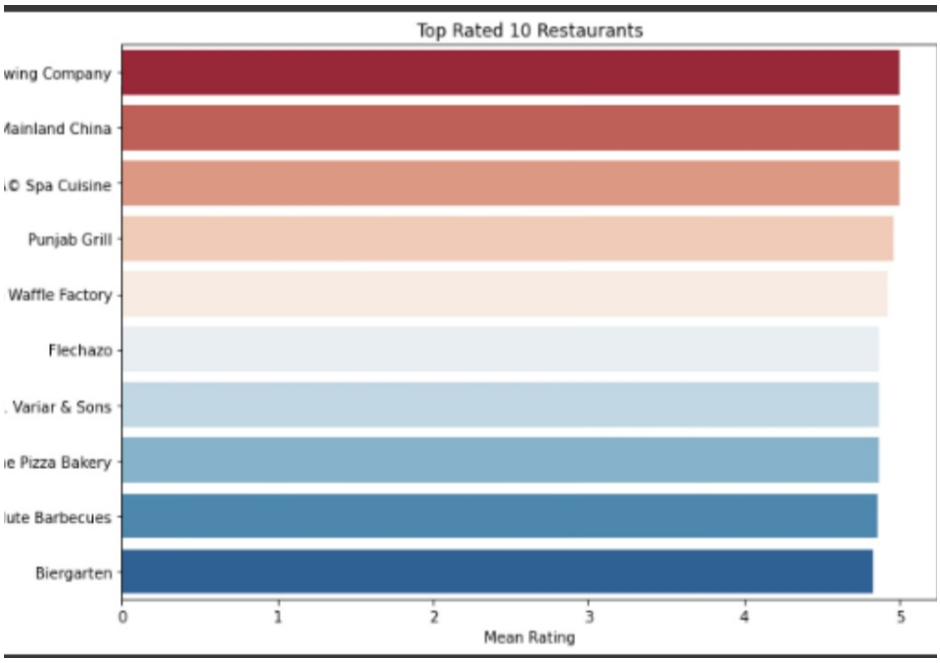


Checking the distribution of restaurant rating, for that we are using distplot from seaborn library

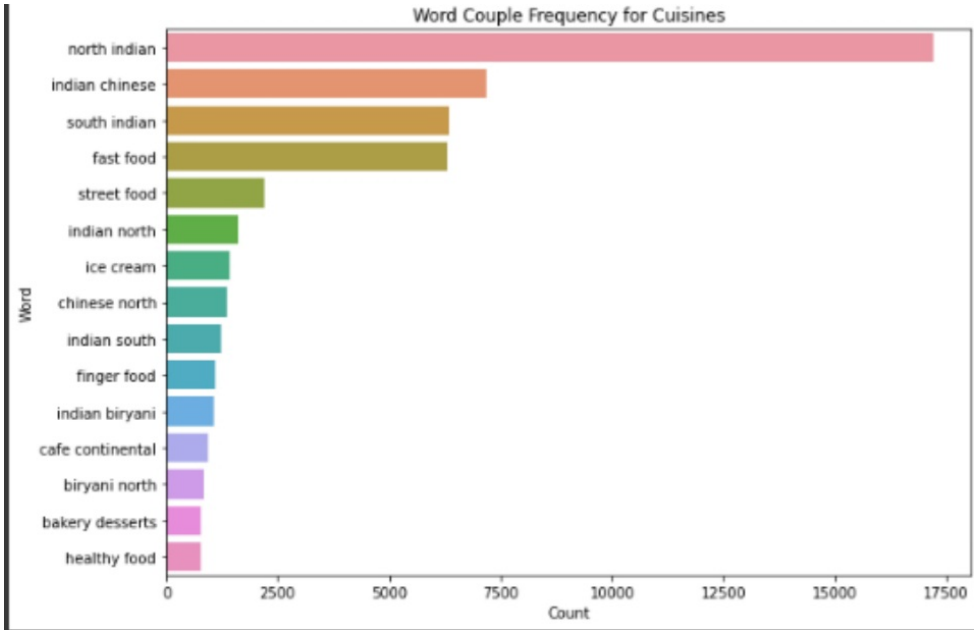


And we can infer that most of the restaurants in Bangalore have rating above 3.5.

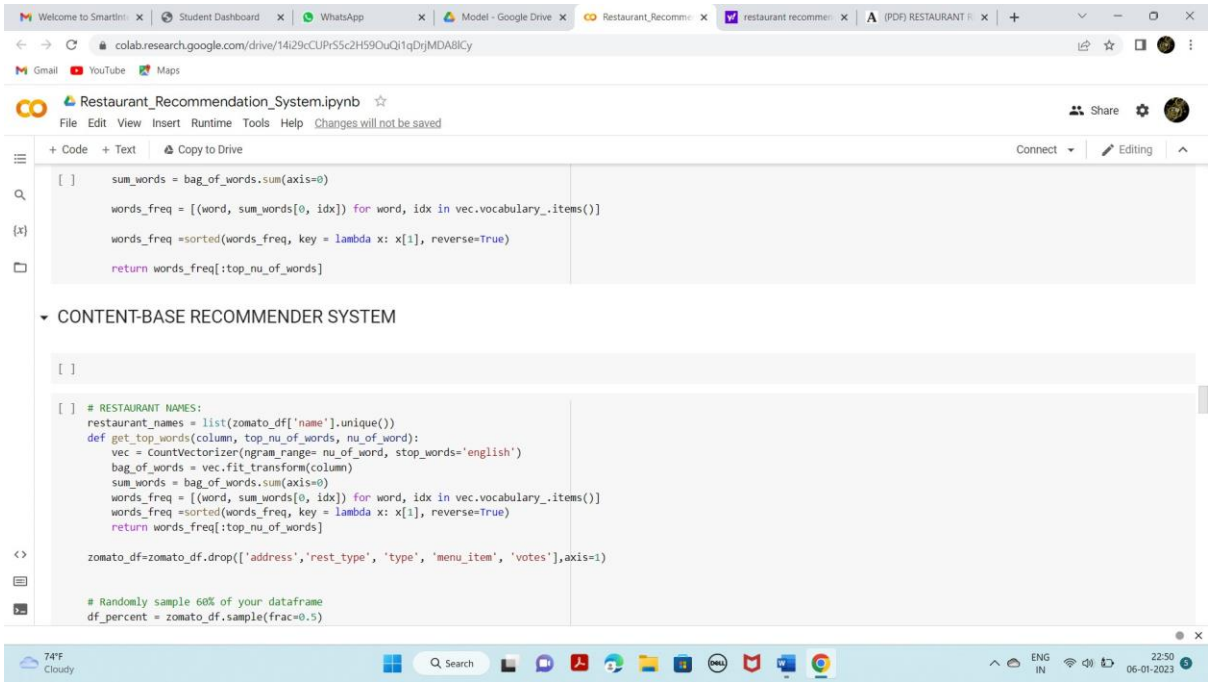
Visualizing top 10 rated restaurants in Bangalore. For that we are again using barplot from Matplotlib library



Visualizing two word frequencies for cuisines, using barplot from seaborn library.



Here we can see the Top favourite cuisine among people of Bangalore is ‘North Indian’, ‘Indian Chinese’ and ‘Fast food’.

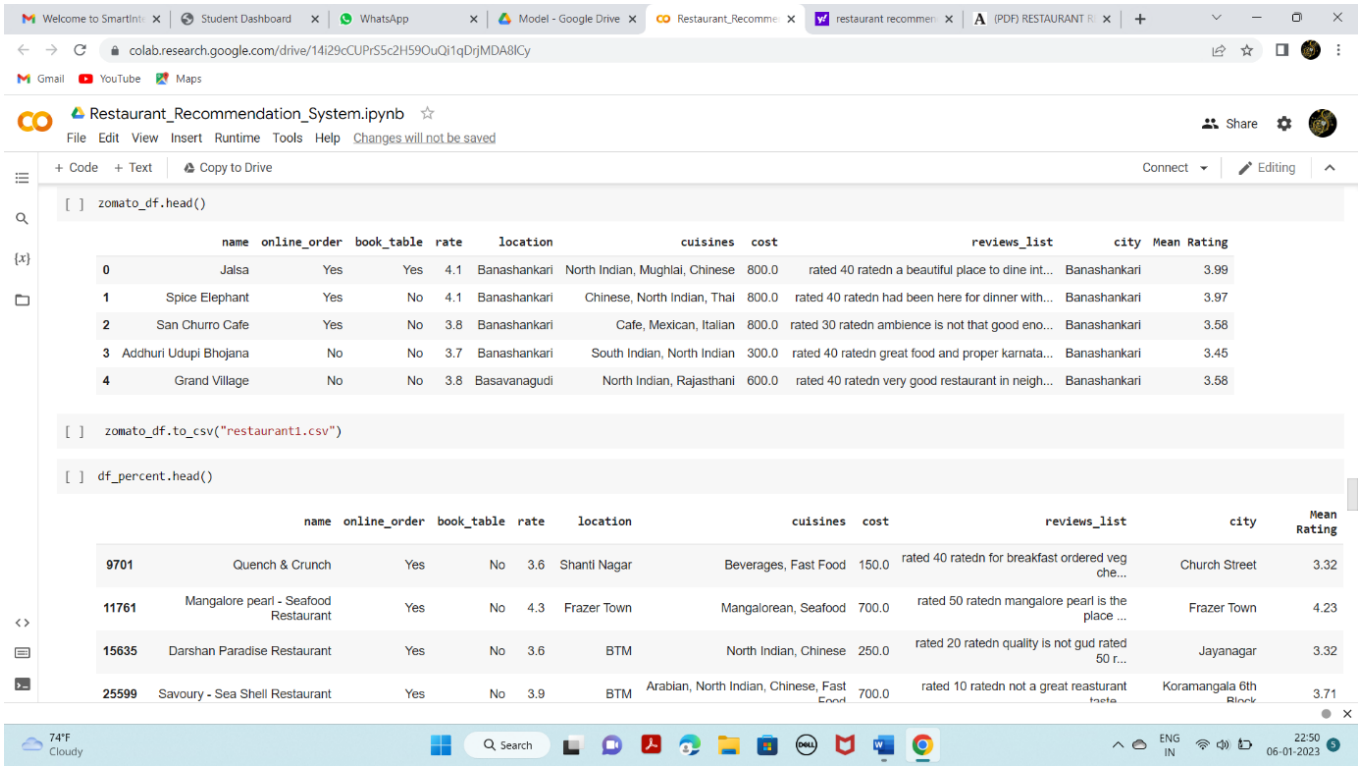


Calculating Cosine Similarity

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

The formula for Cosine Similarity

And in the last line of code, we are calculating the cosine similarity of each item with every other item in the dataset. So we just pass the matrix as an argument.



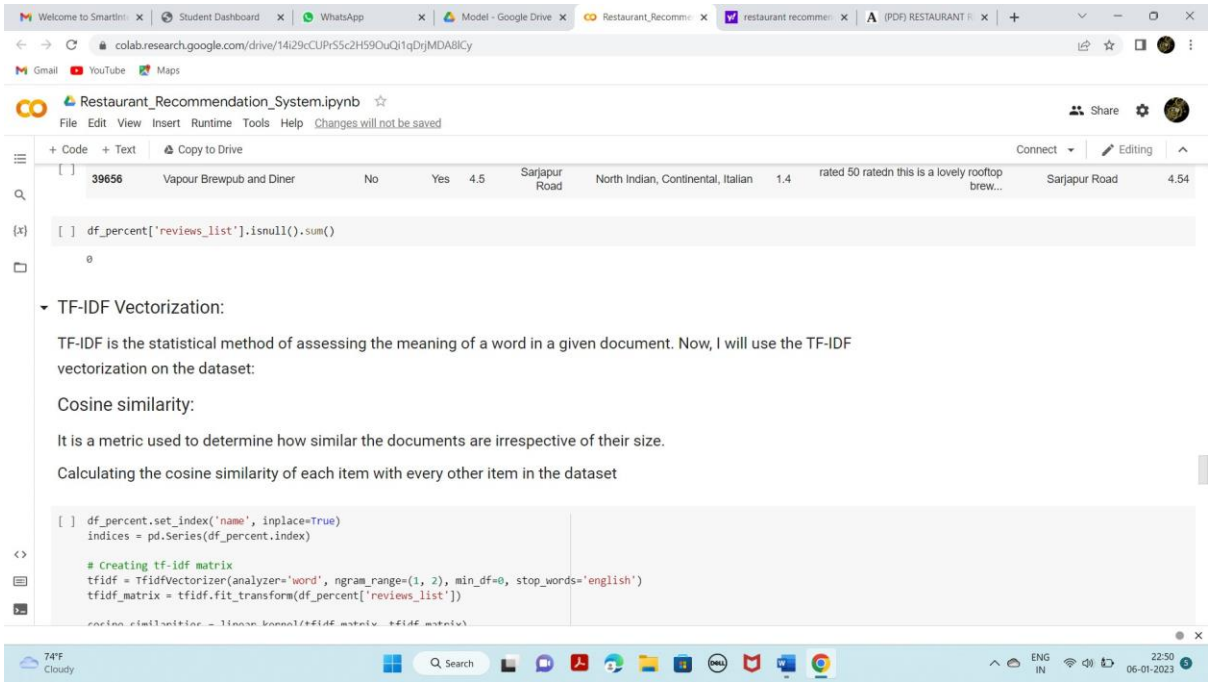
	name	online_order	book_table	rate	location	cuisines	cost	reviews_list	city	Mean Rating
0	Jalsa	Yes	Yes	4.1	Banashankari	North Indian, Mughlai, Chinese	800.0	rated 40 ratedn a beautiful place to dine int...	Banashankari	3.99
1	Spice Elephant	Yes	No	4.1	Banashankari	Chinese, North Indian, Thai	800.0	rated 40 ratedn had been here for dinner with...	Banashankari	3.97
2	San Churro Cafe	Yes	No	3.8	Banashankari	Cafe, Mexican, Italian	800.0	rated 30 ratedn ambience is not that good eno...	Banashankari	3.58
3	Addhuri Udupi Bhojana	No	No	3.7	Banashankari	South Indian, North Indian	300.0	rated 40 ratedn great food and proper karnata...	Banashankari	3.45
4	Grand Village	No	No	3.8	Basavanagudi	North Indian, Rajasthani	600.0	rated 40 ratedn very good restaurant in neigh...	Banashankari	3.58

	name	online_order	book_table	rate	location	cuisines	cost	reviews_list	city	Mean Rating
9701	Quench & Crunch	Yes	No	3.6	Shanti Nagar	Beverages, Fast Food	150.0	rated 40 ratedn for breakfast ordered veg che...	Church Street	3.32
11761	Mangalore pearl - Seafood Restaurant	Yes	No	4.3	Frazer Town	Mangalorean, Seafood	700.0	rated 50 ratedn mangalore pearl is the place ...	Frazer Town	4.23
15635	Darshan Paradise Restaurant	Yes	No	3.6	BTM	North Indian, Chinese	250.0	rated 20 ratedn quality is not gud rated 50 r...	Jayanagar	3.32
25599	Savoury - Sea Shell Restaurant	Yes	No	3.9	BTM	Arabian, North Indian, Chinese, Fast Food	700.0	rated 10 ratedn not a great reasturant taste	Koramangala 6th Block	3.71

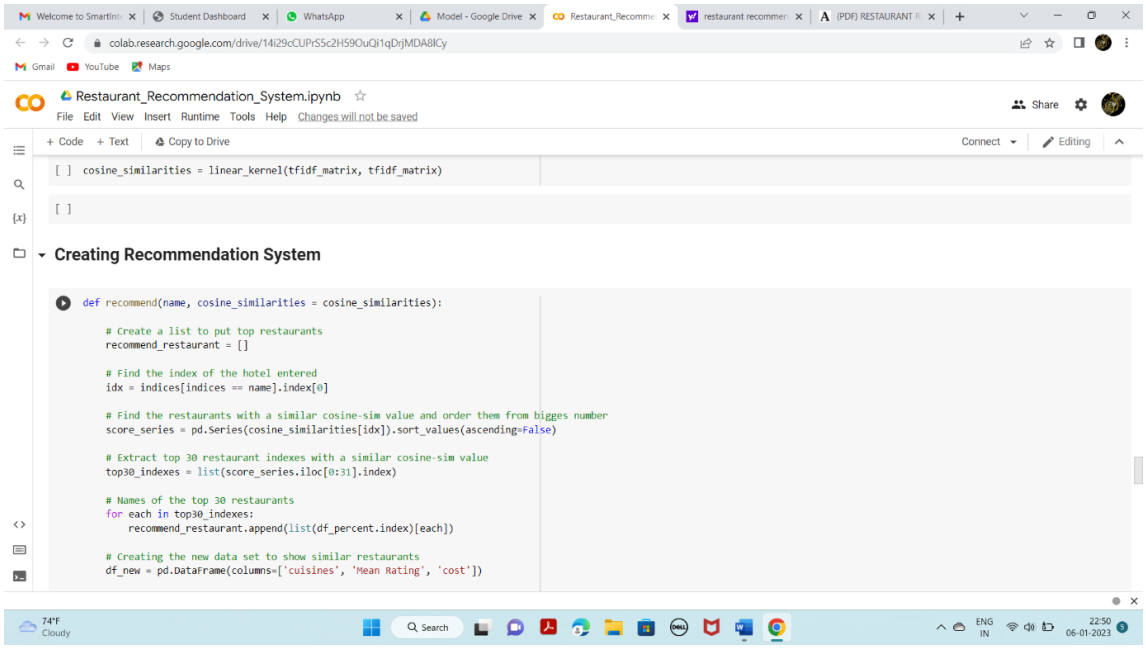
TF-IDF Matrix (Term Frequency — Inverse Document Frequency Matrix)

TF-IDF is a statistical method of assessing the meaning of a word in a given document. Now we use TF-IDF vectorization on the dataset.

Cosine similarity is a metric used to determine how similar the documents are irrespective of their size. Calculating the cosine similarity of each item with every other item in the dataset.



Here, the tf-idf matrix is the matrix containing each word and its TF-IDF score with regard to each document, or item in this case. Also, stop words are simply words that add no significant value to our system, like ‘an’, ‘is’, ‘the’, and hence are ignored by the system.



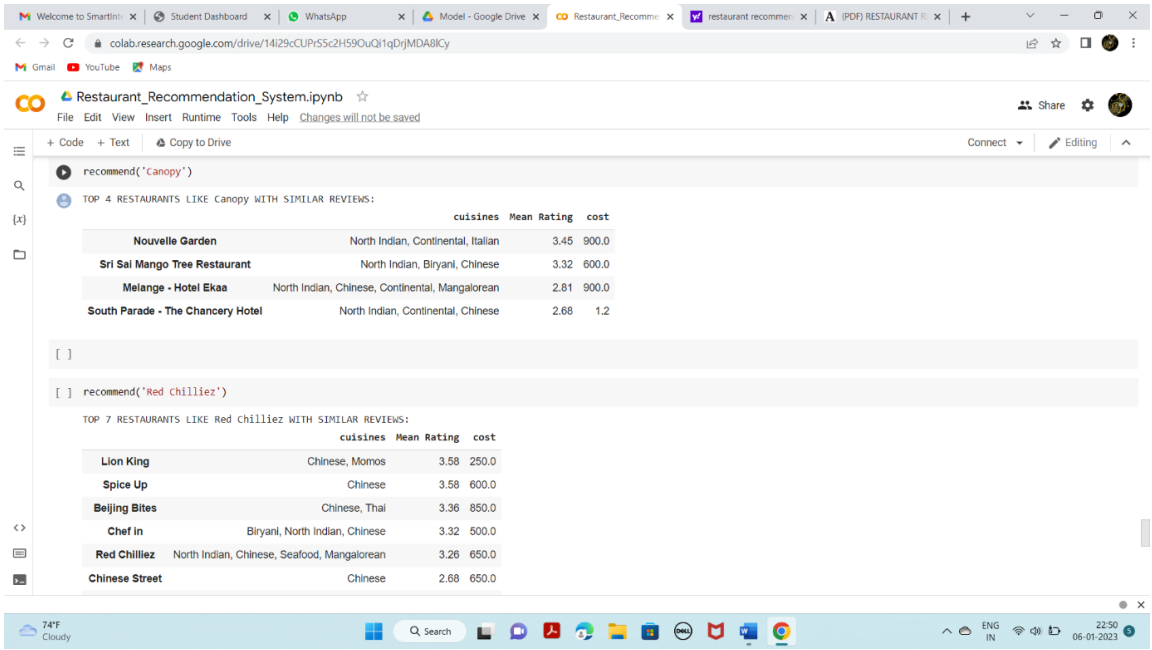
```
[ ] cosine_similarities = linear_kernel(tfidf_matrix, tfidf_matrix)

[ ]
```

Creating Recommendation System

```
def recommend(name, cosine_similarities = cosine_similarities):

    # Create a list to put top restaurants
    recommend_restaurant = []

    # Find the index of the hotel entered
    idx = indices[indices == name].index[0]

    # Find the restaurants with a similar cosine-sim value and order them from bigges number
    score_series = pd.Series(cosine_similarities[idx]).sort_values(ascending=False)

    # Extract top 30 restaurant indexes with a similar cosine-sim value
    top30_indexes = list(score_series.iloc[0:31].index)

    # Names of the top 30 restaurants
    for each in top30_indexes:
        recommend_restaurant.append(list(df_percent.index)[each])

    # Creating the new data set to show similar restaurants
    df_new = pd.DataFrame(columns=['cuisines', 'Mean Rating', 'cost'])
```

Querying recommendation for 4 Restaurants:

For Restaurant ‘Red Chilliez’



```
recommend('Canopy')
```

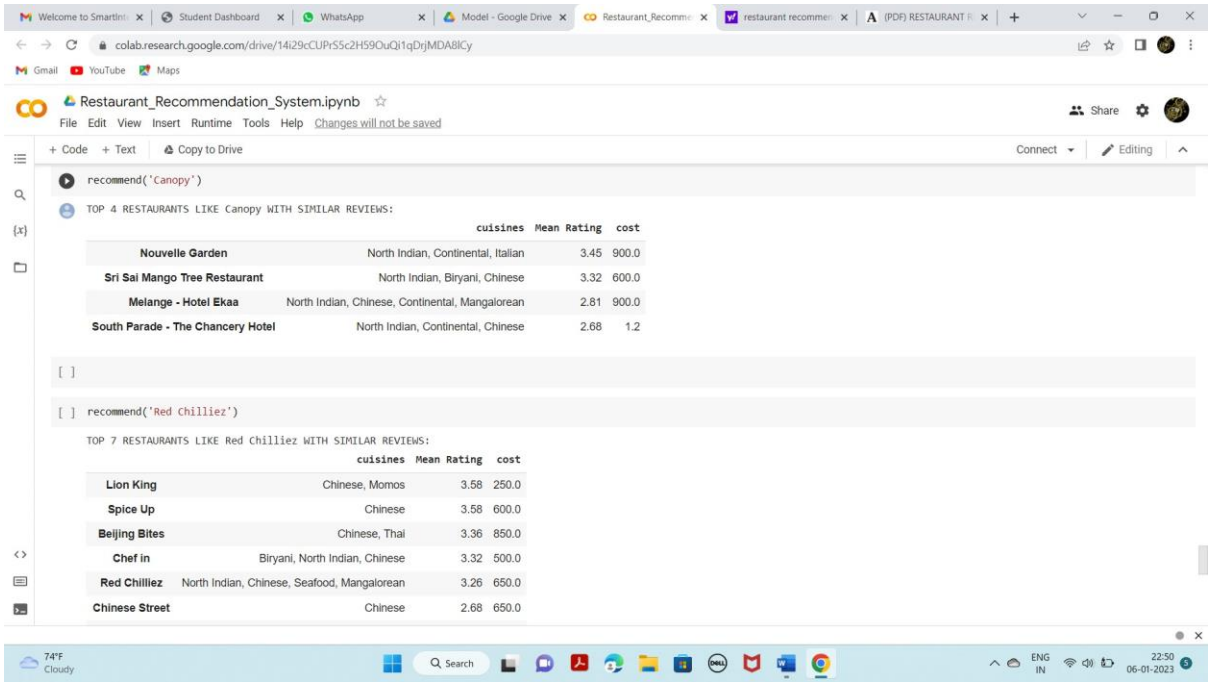
TOP 4 RESTAURANTS LIKE Canopy WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Nouvelle Garden	North Indian, Continental, Italian	3.45	900.0
Sri Sai Mango Tree Restaurant	North Indian, Biryani, Chinese	3.32	600.0
Melange - Hotel Ekaa	North Indian, Chinese, Continental, Mangalorean	2.81	900.0
South Parade - The Chancery Hotel	North Indian, Continental, Chinese	2.68	1.2

```
[ ]

[ ] recommend('Red Chilliez')
```

TOP 7 RESTAURANTS LIKE Red Chilliez WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Lion King	Chinese, Momos	3.58	250.0
Spice Up	Chinese	3.58	600.0
Beijing Bites	Chinese, Thai	3.36	850.0
Chef in	Biryani, North Indian, Chinese	3.32	500.0
Red Chilliez	North Indian, Chinese, Seafood, Mangalorean	3.26	650.0
Chinese Street	Chinese	2.68	650.0



Restaurant_Recommendation_System.ipynb

```
recommend('Canopy')
```

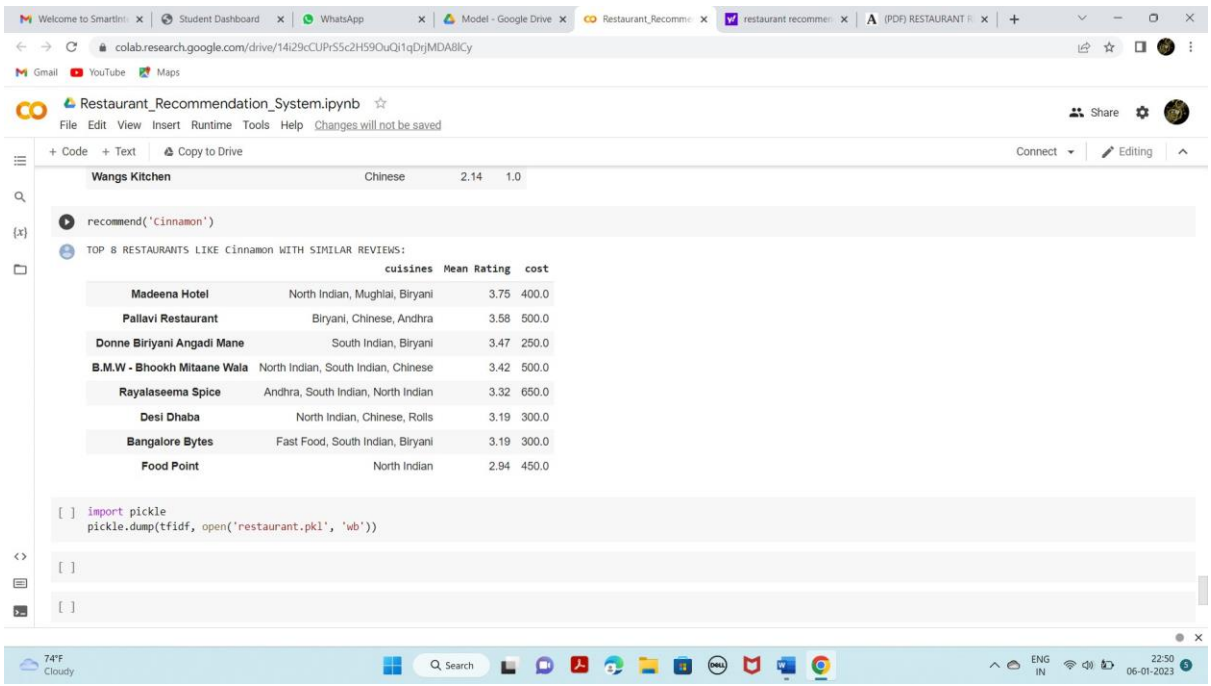
TOP 4 RESTAURANTS LIKE Canopy WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Nouvelle Garden	North Indian, Continental, Italian	3.45	900.0
Sri Sai Mango Tree Restaurant	North Indian, Biryani, Chinese	3.32	600.0
Melange - Hotel Ekaa	North Indian, Chinese, Continental, Mangalorean	2.81	900.0
South Parade - The Chancery Hotel	North Indian, Continental, Chinese	2.68	1.2

```
recommend('Red Chilliez')
```

TOP 7 RESTAURANTS LIKE Red Chilliez WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Lion King	Chinese, Momos	3.58	250.0
Spice Up	Chinese	3.58	600.0
Beijing Bites	Chinese, Thai	3.36	850.0
Chef in	Biryani, North Indian, Chinese	3.32	500.0
Red Chilliez	North Indian, Chinese, Seafood, Mangalorean	3.26	650.0
Chinese Street	Chinese	2.68	650.0



Restaurant_Recommendation_System.ipynb

Wangs Kitchen	Chinese	2.14	1.0

```
recommend('Cinnamon')
```

TOP 8 RESTAURANTS LIKE Cinnamon WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Madeena Hotel	North Indian, Mughlai, Biryani	3.75	400.0
Pallavi Restaurant	Biryani, Chinese, Andhra	3.58	500.0
Donne Biriyani Angadi Mane	South Indian, Biryani	3.47	250.0
B.M.W - Bhookh Mitaane Wala	North Indian, South Indian, Chinese	3.42	500.0
Rayalaseema Spice	Andhra, South Indian, North Indian	3.32	650.0
Desi Dhaba	North Indian, Chinese, Rolls	3.19	300.0
Bangalore Bytes	Fast Food, South Indian, Biryani	3.19	300.0
Food Point	North Indian	2.94	450.0

```
import pickle
pickle.dump(tfidf, open('restaurant.pkl', 'wb'))
```

2.2 .HTML CODE AND PYTHON CODE

app.py

```
import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt
import plotly.offline as py
import plotly.graph_objs as go
import seaborn as sns
import warnings
warnings.filterwarnings('always')
warnings.filterwarnings('ignore')
import nltk
from nltk.corpus import stopwords
from sklearn.metrics.pairwise import linear_kernel
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
import flask
from flask import Flask,render_template, request
import pickle


app = Flask(__name__) # initializing a flask app
model=pickle.load(open("restaurant.pkl",'rb'))  #loading the model

#loading the updated dataset
zomato_df=pd.read_csv("restaurant1.csv")

@app.route('/')# route to display the home page
def home():
    return render_template('home.html')#rendering the home page


@app.route('/extractor')
def extractor():
    return render_template('extractor.html')

#extractor page
@app.route('/keywords',  methods=['POST'])# route to show the
predictions in a web UI
def keywords():
    #typ=request.form['type']
    output=request.form['output']
    #if typ=="text":
        #output=re.sub("[^a-zA-Z.,]"," ",output)
    print(output)
    print(type(output))
    df_percent = zomato_df.sample(frac=0.5)
    df_percent.set_index('name', inplace=True)
    indices = pd.Series(df_percent.index)
    # Creating tf-idf matrix
```

```
    tfidf = TfidfVectorizer(analyzer='word', ngram_range=(1, 2),
min_df=0, stop_words='english')

    tfidf_matrix =
tfidf.fit_transform(df_percent['reviews_list'].fillna(' '))
    cosine_similarities = linear_kernel(tfidf_matrix, tfidf_matrix)

    def recommend(name, cosine_similarities = cosine_similarities):

        # Create a list to put top restaurants
        recommend_restaurant = []

        # Find the index of the hotel entered
        idx = indices[indices == name].index[0]

        # Find the restaurants with a similar cosine-sim value and
order them from bigges number
        score_series =
pd.Series(cosine_similarities[idx]).sort_values(ascending=False)

        # Extract top 30 restaurant indexes with a similar cosine-sim
value
        top30_indexes = list(score_series.iloc[0:31].index)

        # Names of the top 30 restaurants
        for each in top30_indexes:
            recommend_restaurant.append(list(df_percent.index)[each])

        # Creating the new data set to show similar restaurants
        df_new = pd.DataFrame(columns=['cuisines', 'Mean Rating',
'cost'])

        # Create the top 30 similar restaurants with some of their
columns
        for each in recommend_restaurant:
           df_new =
df_new.append(pd.DataFrame(df_percent[['cuisines','Mean Rating',
'cost']][df_percent.index == each].sample()))

        # Drop the same named restaurants and sort only the top 10 by
the highest rating
        df_new = df_new.drop_duplicates(subset=['cuisines','Mean
Rating', 'cost'], keep=False)
        pd.set_option('display.max_columns', None)

        df_new = df_new.sort_values(by='Mean Rating',
ascending=False).head(10)
        print('TOP %s RESTAURANTS LIKE %s WITH SIMILAR REVIEWS: ' %
(str(len(df_new)), name))

        return df_new
```

```
    result = recommend(output)
    print(result)
    print(type(result))



#print(result[0])

    #print(result[0])
   # res = result.to_string(index=False)



    #showing the prediction results in a UI
    return render_template('keywords.html',keyword=result.to_html())


if __name__ == "__main__":
   # running the app
    app.run(debug=False)
```

3.RESULTS

This is the home main page that describes the project and summarizes it.

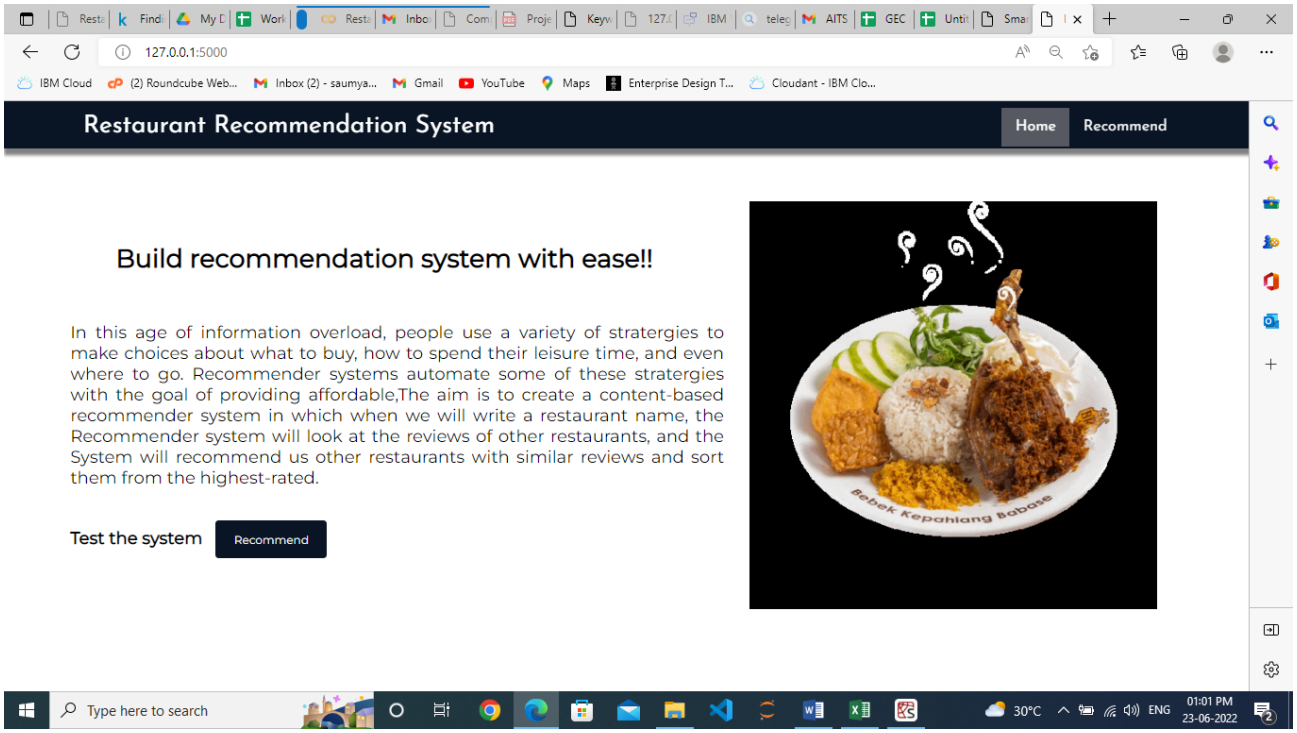


Figure 1: Home Page(Which gives introduction to Restaurant Recommendation system)

Checking recommendation for the restaurant: ‘Jalsa’

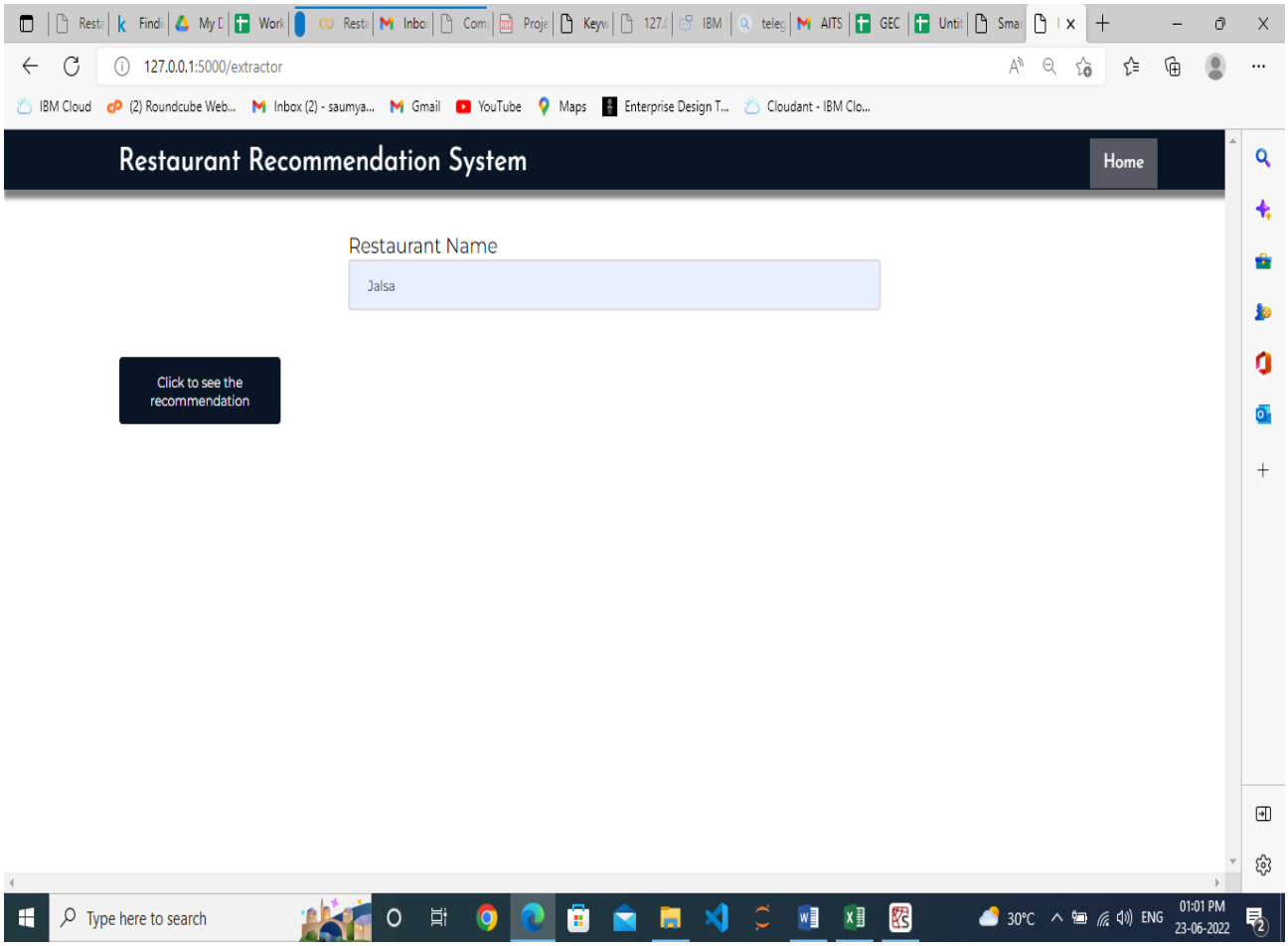
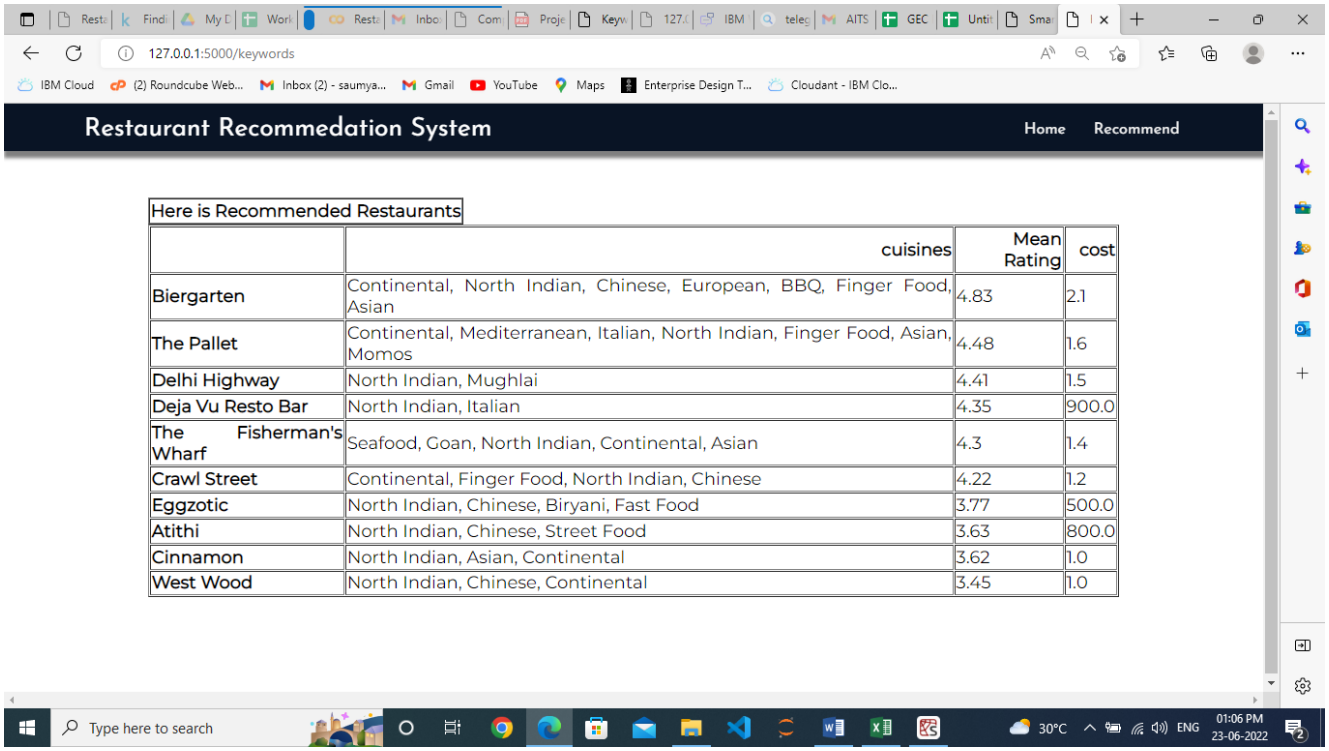


Figure 2:Input pages(Which takes inputs from User)



	cuisines	Mean Rating	cost
Biergarten	Continental, North Indian, Chinese, European, BBQ, Finger Food, Asian	4.83	2.1
The Pallet	Continental, Mediterranean, Italian, North Indian, Finger Food, Asian, Momos	4.48	1.6
Delhi Highway	North Indian, Mughlai	4.41	1.5
Deja Vu Resto Bar	North Indian, Italian	4.35	900.0
The Fisherman's Wharf	Seafood, Goan, North Indian, Continental, Asian	4.3	1.4
Crawl Street	Continental, Finger Food, North Indian, Chinese	4.22	1.2
Eggzotic	North Indian, Chinese, Biryani, Fast Food	3.77	500.0
Atithi	North Indian, Chinese, Street Food	3.63	800.0
Cinnamon	North Indian, Asian, Continental	3.62	1.0
West Wood	North Indian, Chinese, Continental	3.45	1.0

This is the prediction page where we will provide a restaurant name for which we will get the top recommended restaurants, which based on cuisines, mean rating (out of 5), cost in thousands.

Figure 3: Output Page (Display the Restaurant name and Details)

Checking recommendation for the restaurant ‘Cinnamon’

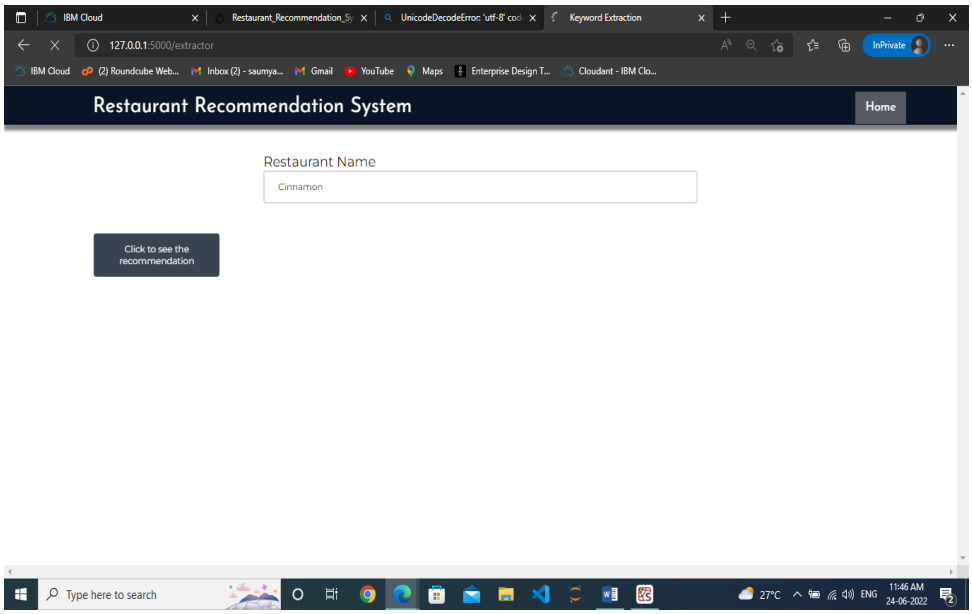
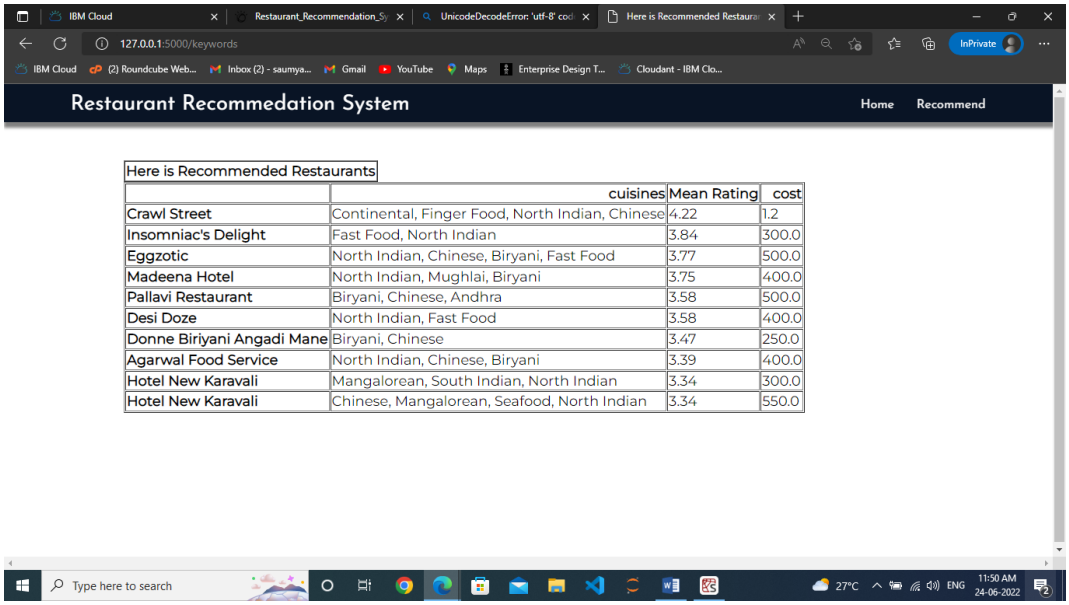


Figure 4: Home Page (Which gives introduction to Restaurant Recommendation system)



Here is Recommended Restaurants

	cuisines	Mean Rating	cost
Crawl Street	Continental, Finger Food, North Indian, Chinese	4.22	1.2
Insomniac's Delight	Fast Food, North Indian	3.84	300.0
Eggzotic	North Indian, Chinese, Biryani, Fast Food	3.77	500.0
Madeena Hotel	North Indian, Mughlai, Biryani	3.75	400.0
Pallavi Restaurant	Biryani, Chinese, Andhra	3.58	500.0
Desi Doze	North Indian, Fast Food	3.58	400.0
Donne Biriyani Angadi Mane	Biryani, Chinese	3.47	250.0
Agarwal Food Service	North Indian, Chinese, Biryani	3.39	400.0
Hotel New Karavali	Mangalorean, South Indian, North Indian	3.34	300.0
Hotel New Karavali	Chinese, Mangalorean, Seafood, North Indian	3.34	550.0

Finally, the prediction for the given restaurant inputs is shown.

Figure 5: Output Page (Display the Restaurant name and Details)

4.APPLICATIONS

- Can be used by tourists to find best reastaurant in new places.

5.ADVANTAGES

Advantages:

- You can try new foods.
- You do not have to cook.
- You get to spend time with family and friends.
- It's easier to feed large parties.
- Time is not wasted.

6.DISADVANTAGES

Disadvantages:

- Significant investments required.
- Too many choices.
- The complex onboarding process.
- Lack of data analytics capability.
- The 'cold start' problem.
- Inability to capture changes in user behavior.
- Privacy concerns.

7.FUTURE SCOPE

- To build more friendly graphical interfaces, the next goal for a further project is to improve the performance of system and improve the member's benefit such as providing more functions for member, like online reservation function, online order menu, and website, etc

8. BIBILOGRAPHY

https://medium.com/mlearning-ai/restaurant-recommendation-system-based-on-the-content-in-reviews-dfc3351004db

https://www.kaggle.com/code/midouazerty/restaurant-recommendation-system-using-ml

https://towardsdatascience.com/yelp-restaurant-recommendation-system-capstone-project-264fe7a7dea1

https://www.academia.edu/85069823/RESTAURANT_RECOMMENDATION_SYSTEM

9.HELP FILE

PROJECT EXECUTION:

STEP-1: Go to Start, search and launch ANACONDA NAVIGATOR.

STEP-2: After launching of ANACONDA NAVIGATOR, launch JUPYTER NOTEBOOK.

STEP-3: Open “Major project code” IPYNB file.

STEP-4: Then run all the cells.

STEP-5: All the data preprocessing, training and testing, model building, accuracy of the model can be showcased.

STEP-6: And a pickle file will be generated.

STEP-7: Create a Folder named FLASK on the DESKTOP. Extract the pickle file into this Flask Folder.

STEP-8: Extract all the html files (home.html, index.html, chance.html, nochance.html) and python file(app.py) into the FLASK Folder.

STEP-9: Then go back to ANACONDA NAVIGATOR and the launch the SPYDER.

STEP-10: After launching Spyder, give the path of FLASK FOLDER which you have

created on the DESKTOP.

STEP-11: Open all the app.py and html files present in the Flask Folder.

STEP-12: After running of the app.py, open ANACONDA PROMPT and follow the below steps: cd File Path□click enter

python app.py□click enter (We could see running of files).

STEP-13: Then open BROWSER, at the URL area type ―localhost:5000”.

STEP-14: Home page of the project will be displayed.

STEP-15: Click on ―Go to Predict”. Directly it will be navigated to index page.

STEP-16: A index page will be displayed where the user needs to give the inputs and then

click on ―” Predict”. Output will be famous restaurant in the particular area.