INSTITUTE FOR ADVANCED COMPUTING AND SOFTWARE DEVELOPMENT

AKURDI, PUNE

Documentation On

"Zomato Restaurant Recommendation System"

PG-DBDA SEP 2020

Submitted By:
Group No: 17

Kanchan Mahajan – 1524

Shrutika Makhamale – 1542

Project Guide: **Mr. Prashant Karhale**

Mr. Akshay Tilekar (External Guide) **Centre Coordinator**

Mr. Rahul Pund (Internal Guide)

Mr. Manish Bendale (Internal Guide)

INDEX

Acknowledgement

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organisations. We would like to extend our sincere thanks to all of them. We are highly indebted to Mr. Prashant Karhale, Centre Co-Ordinator IACSD CDAC, Akurdi Pune for constant supervision as well as for providing the opportunity.

We would like to express our gratitude towards our faculties, project Guide Mr. Akshay Tilekar, Internal Project Guides Mr. Rahul , Mr. Manish and Mrs. Tejal, for their kind co-operation and encourage-ment which help us in completion of this project. Our thanks and appreciations also go to the colleagues in developing the project and people who have willingly helped us out with their abilities.

Abstract

Restaurant recommendation system is a very popular service whose sophistication keeps increasing everyday. In this paper we present a personalised restaurant recommendation system which has two parts to it. The first part recommends users' restaurants based on their restaurant review history. The second part recommends business owners with places perfect to open a restaurant with a particular cuisine where the owner would get the best traffic for the restaurant. Using Zomato data, we built a restaurant recommendation system for the individuals and business owners. For each user in our data we find out the cuisine preferences and other restrictions such as services offered, etc. and based on that we recommend the restaurants accordingly. We propose a metric that takes the popularity as well as the sentiment of opinions for the food items based on the user generated reviews as opposed to other systems where which only consider the features mentioned above to recommend restaurants.

Chapter 1

1.1 Introduction:

What are Recommendation Systems?

Recommender systems have changed the way people find products, information, and even other people. The goal of a Recommender System is to generate meaningful recommendations to a collection of users for items or products that might interest them. It has changed the way inanimate websites communicate with their users. Rather than providing a static experience in which users search for and potentially buy products, recommender systems increase interaction to provide a richer experience. The systems identify recommendations autonomously for individual users based on past purchases and searches, and on other users' behavior. They study patterns of behavior to know what someone will prefer from among a collection of things he has never experienced. The technology behind recommender systems has evolved over the past 20 years into a rich collection of tools that enable the practitioner or researcher to develop effective recommenders.

Collaborative Filtering

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users behaviors, activities or preferences and predicting what users will like based on their similarity to other users. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an understanding of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems. For example, the k-nearest neighbor (k-NN) approach and the Pearson Correlation.

Content Based Filtering

Content-based filtering methods are based on a description of the item and a profile of the users preference. In a content-based recommendation system, keywords are used to describe the items; beside, a user profile is built to indicate the type of item this user likes. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past (or is examining in the present). In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. This approach has its roots in information retrieval and information filtering research.

1.2 Purpose :

Zomato is one of such an app which gives the users the ratings and the reviews on restaurants all over India.Ratings or the Reviews are considered as one of the most important/deciding factors which determine how good a restaurant is.Hence here in our project we will use the real time Data set which has various factors/ features a user would look into regarding a restaurant.We are limiting our data to Bangalore City only. The Zomato dataset gives us information about factors affecting the establishment of different types of restaurant at different places in Bengaluru, aggregate rating of each restaurant.This dataset has 51717 rows and 17 columns.Here we would aim at nding the best cheapest Restaurant in bengaluru. Along with the same we can explore various other relationships like the best expensive restaurant,Best Location,Relation between Location and Rating,No. of restaurants in a Location all these relationships could be visualized.

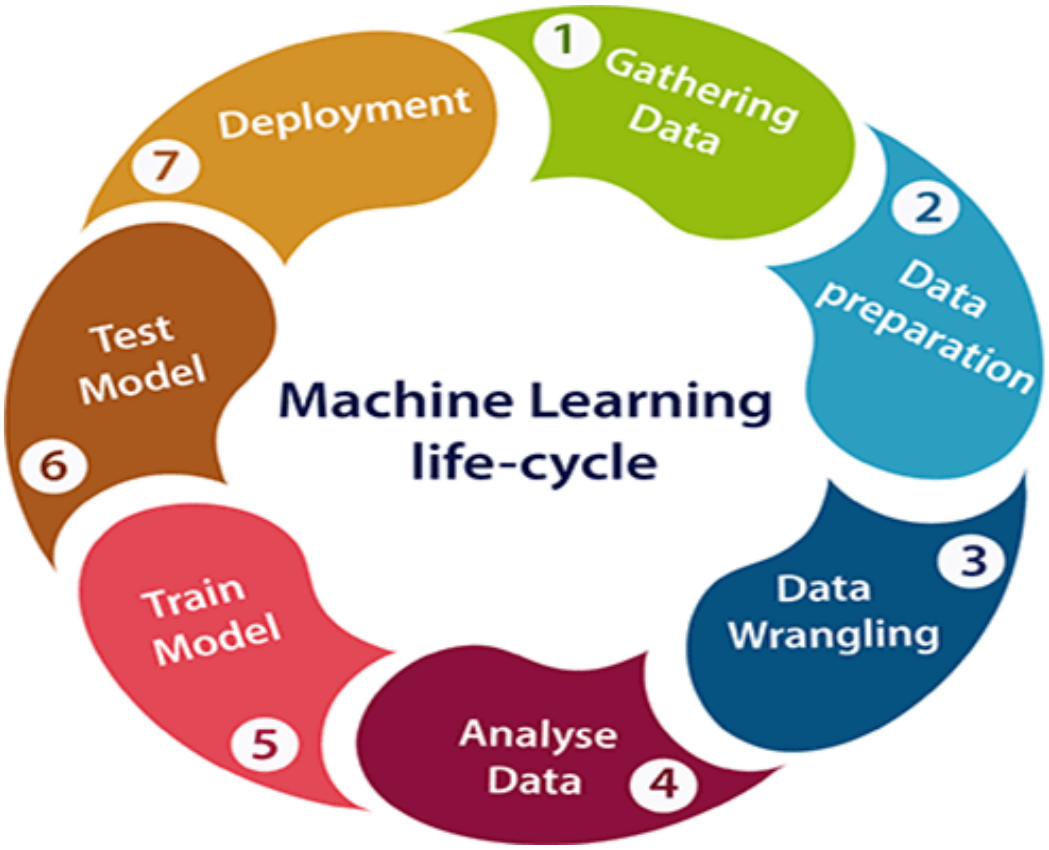
1.3 Scope of the project :

There are many recommendation systems available for problems like shopping,online video entertainment,games etc.Restaurants & Dining is one area where there is a big opportuninty to recommend dining options to users based on their Preferences as well as historical data. Zomato is a very good source of such data with hot only restaurant reviews,but also user-level information on their preferred restaurants. This report describes the work to recommend restaurants to a given Zomato user based on their history or their cuisine preference, It also does the task of recommending cuisine specific suitable locations to newcomers in the restaurant business.

Chapter 2

2.1 Machine learning Life cycle

Machine learning has given the computer systems the abilities to automatically learn without being explicitly programmed. But how does a machine learning system work? So, it can be described using the life cycle of machine learning. Machine learning life cycle is a cyclic process to build an efficient machine learning project. The main purpose of the life cycle is to find a solution to the problem or project.



The most important thing in the complete process is to understand the problem and to know the purpose of the problem. Therefore, before starting the life cycle, we need to understand the problem because the good result depends on the better understanding of the problem.
In the complete life cycle process, to solve a problem, we create a machine learning system called "model", and this model is created by providing "training". But to train a model, we need data, hence, life cycle starts by collecting data.

Machine learning life cycle involves seven major steps, which are given below:

- **Gathering Data**
- **Data preparation**
- **Data Wrangling**
- **Analyse Data**
- **Train the model**
- **Test the model**
- **Deployment**

1. Gathering Data:

Data Gathering is the first step of the machine learning life cycle. The goal of this step is to identify and obtain all data-related problems.

In this step, we need to identify the different data sources, as data can be collected from kaggle. It is one of the most important steps of the life cycle. The quantity and quality of the collected data will determine the efficiency of the output. The more will be the data, the more accurate will be the prediction.

This step includes the below tasks:

- **Identify various data sources**
- **Collect data**

By performing the above task, we get a coherent set of data, also called as a **dataset**. It will be used in further steps.

2. Data preparation

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.

In this step, first, we put all data together, and then randomize the ordering of data.

This step can be further divided into two processes:

- **Data exploration:**
  It is used to understand the nature of data that we have to work with. We need to understand the characteristics, format, and quality of data.
  A better understanding of data leads to an effective outcome. In this, we find Correlations, general trends, and outliers.
- **Data pre-processing:**
  Now the next step is preprocessing of data for its analysis.

3. Data Wrangling

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:

- **Missing Values**
- **Duplicate data**
- **Invalid data**
- **Noise**

So, we use various filtering techniques to clean the data.

It is mandatory to detect and remove the above issues because it can negatively affect the quality of the outcome.

4. Data Analysis

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- **Selection of analytical techniques**
- **Building models**
- **Review the result**

The aim of this step is to build a machine learning model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as KNN,TF-IDF etc. then build the model using prepared data, and evaluate the model.

Hence, in this step, we take the data and use machine learning algorithms to build the model.

5. Train Model

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.

We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and, features.

6. Test Model

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.

Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

7. Deployment

The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

2.2 Overall Description

2.2.1 Data:

https://www.kaggle.com/himanshupoddar/zomato-bangalore-restaurants

```
1 zomato_data=pd.read_csv("/content/drive/MyDrive/dataset/zomato.csv")
2 zomato_data.head()
```

	url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_liked	cuisines	approx_cost(for two people)	reviews_list	menu_item	listed_in(type)	listed_in
	ngalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1/5	775	080 42297555\r\n+91 9743772233	Banashankari	Casual Dining	Pasta, Lunch Buffet, Masala Papad, Paneer Laja...	North Indian, Mughlai, Chinese	800	[('Rated 4.0', 'RATED\n A beautiful place to ...	[]	Buffet	Banas
	ngalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1/5	787	080 41714161	Banashankari	Casual Dining	Momos, Lunch Buffet, Chocolate Nirvana, Thai G...	Chinese, North Indian, Thai	800	[('Rated 4.0', 'RATED\n Had been here for din...	[]	Buffet	Banas
	roBangalore? cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8/5	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churros, Cannelloni, Minestrone Soup, Hot Choc...	Cafe, Mexican, Italian	800	[('Rated 3.0', 'RATED\n Ambience is not that ...	[]	Buffet	Banas
	alore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	No	3.7/5	88	+91 9620009302	Banashankari	Quick Bites	Masala Dosa	South Indian, North Indian	300	[('Rated 4.0', 'RATED\n Great food and proper...	[]	Buffet	Banas
	igalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8/5	166	+91 8026612447\r\n+91 9901210005	Basavanagudi	Casual Dining	Panipuri, Gol Gappe	North Indian, Rajasthani	600	[('Rated 4.0', 'RATED\n Very good restaurant ...	[]	Buffet	Banas

The dataset contains the following features :

1. *url* : This feature contains the url of the restaurant on the Zomato website
2. *address* : This feature contains the address of the restaurant in Bangalore
3. *name* : This feature contains the name of the restaurant
4. *online_order* : whether online ordering is available in the restaurant or not
5. *book_table* : table book option available or not
6. *rate* : contains the overall rating of the restaurant out of 5
7. *votes* : contains total number of upvotes for the restaurant

8. *phone* : contains the phone number of the restaurant
9. location : contains the neighborhood in which the restaurant is located
10. *rest_type* : restaurant type
11. *dish_liked* : dishes people liked in the restaurant
12. *cuisines* : food styles, separated by comma
13. *approx_cost(for two people)* : contains the approximate cost of meal for two people
14. *reviews_list* : list of tuples containing reviews for the restaurant, each tuple consists of two values, rating and review by the customer
15. *menu_item* : contains list of menus available in the restaurant
16. *listed_in(type)* : type of meal
17. *listed_in(city)* : contains the neighborhood in which the restaurant is located

The dataset contains 51717 rows.

2.2.2 Imports:

- **Matplotlib:**

Matplotlib is a collection of command style functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure. We have used it to show visualizations of analysis.

- **Numpy:**

Numpy is used to for mathematical operations. This package provides easy use of mathematical function

- **Pandas :**

Pandas is an open-source python package built on top of Numpy developed by Wes McKinney**.** It is used as one of the most important data cleaning and analysis tool. It provides fast, flexible, and expressive data structures.

- **Seaborn :**

Seaborn is a plotting library that offers a simpler interface, sensible defaults for plots needed for machine learning, and most importantly, the plots are aesthetically better looking than those in Matplotlib.

2.3 Requirement Specification

2.3.1 Hardware Requirement:

- Processor: Intel Dual Core and above
- RAM: Minimum 8GB
- OS: Windows

2.3.2 Software Requirement:

- Colab
- Jupyter NoteBook

Chapter 3

3.1 Plan of Project Execution

3.1.1 Data Cleaning:

Lets start by deleting the unnecessary or redundant features.

For data analysis, we do not need the contact details of the restaurant so, deleting the following features :
url, *address* and *phone*.

Now, let us consider the *menu_item* and *reviews_list* features.
menu_item feature contains the names of the dishes available in the restaurant. This will hardly impact our analysis, as it is highly mathematics driven. Also, we have other features like *rest_type*, *cuisines*, *listed_in(type)* and *dish_liked* which give us a fair idea about what the restaurants have to offer, as we do not need to be very specific about what all dishes are available in the restaurant for the analysis. So, we will drop this feature as well.

Moving on to *reviews_list* feature. This feature contains the reviews for the restaurant, that were available on Zomato(Bangalore) website.
This feature also will hardly contribute for the analysis as it is pure text and we also have features like *rate* and *votes* that cover the necessary information.
But, one can still extract this feature and play around like running an NLP algorithm and …(you get the idea, right).
As far as the data analysis is considered I will delete this feature as well.

Now that we have deleted some of the features lets clean the remaining.
We see that the *rate* feature is a string as it contains the character '/'.
This is not required, and so I will remove the '/5' from each entry in the column, changing the datatype of the feature to float.

Next let us rename some features, for simplicity sake.
The following features were renamed :
approx_cost(for two people) : avg_cost
listed_in(type) : meal_type
listed_in(city) : city

The feature description given on kaggle, makes one think about the two features : *location* and *city*. Both have the same description, so this is redundant information.
Now, which one to keep?
In this approach I kept the city feature(0 null values), this is my personal preference and would highly recommend to analyze for the other feature as well. So, deleting the *location* feature.

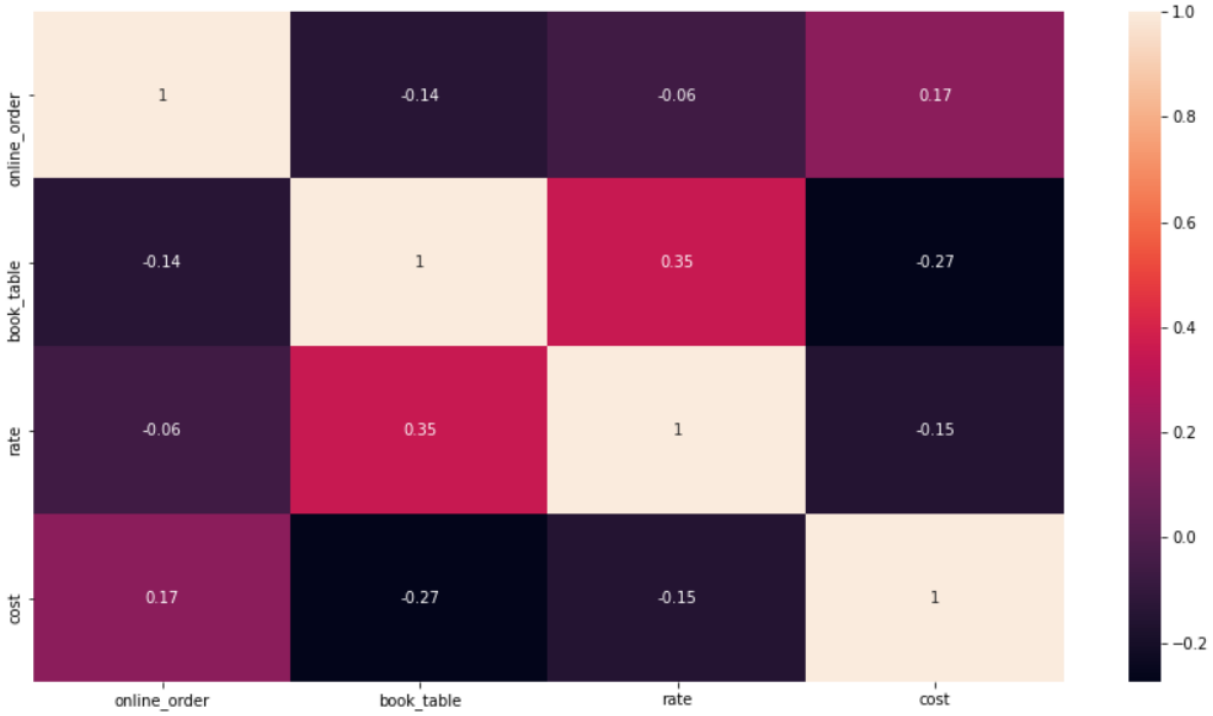
```
1 zomato_data.isnull().sum()
```

```
url                            0
address                        0
name                           0
online_order                   0
book_table                     0
rate                        7775
votes                          0
phone                       1208
location                      21
rest_type                    227
dish_liked                 28078
cuisines                      45
approx_cost(for two people)  346
reviews_list                   0
menu_item                      0
listed_in(type)                0
listed_in(city)                0
dtype: int64
```

3.1.2 : Visualizing the data:

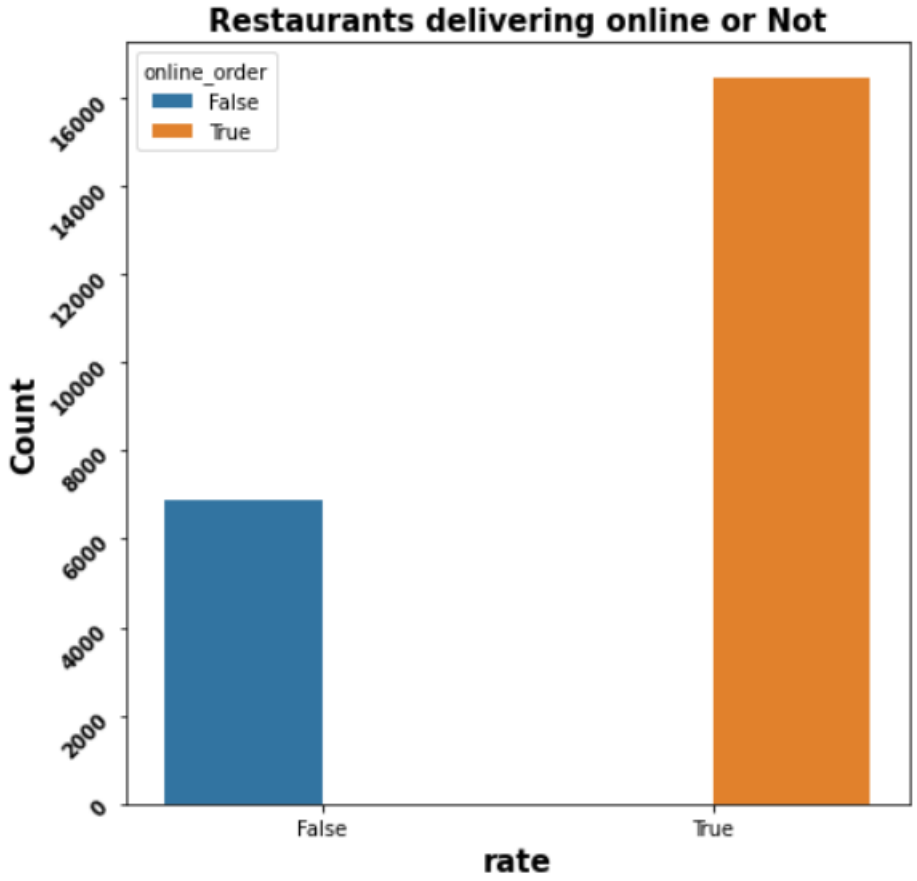
Now let us plot and visualize the features..

```
1 corr = zomato.corr(method='kendall')
2 plt.figure(figsize=(15,8))
3 sns.heatmap(corr, annot=True)
4 zomato.columns
```



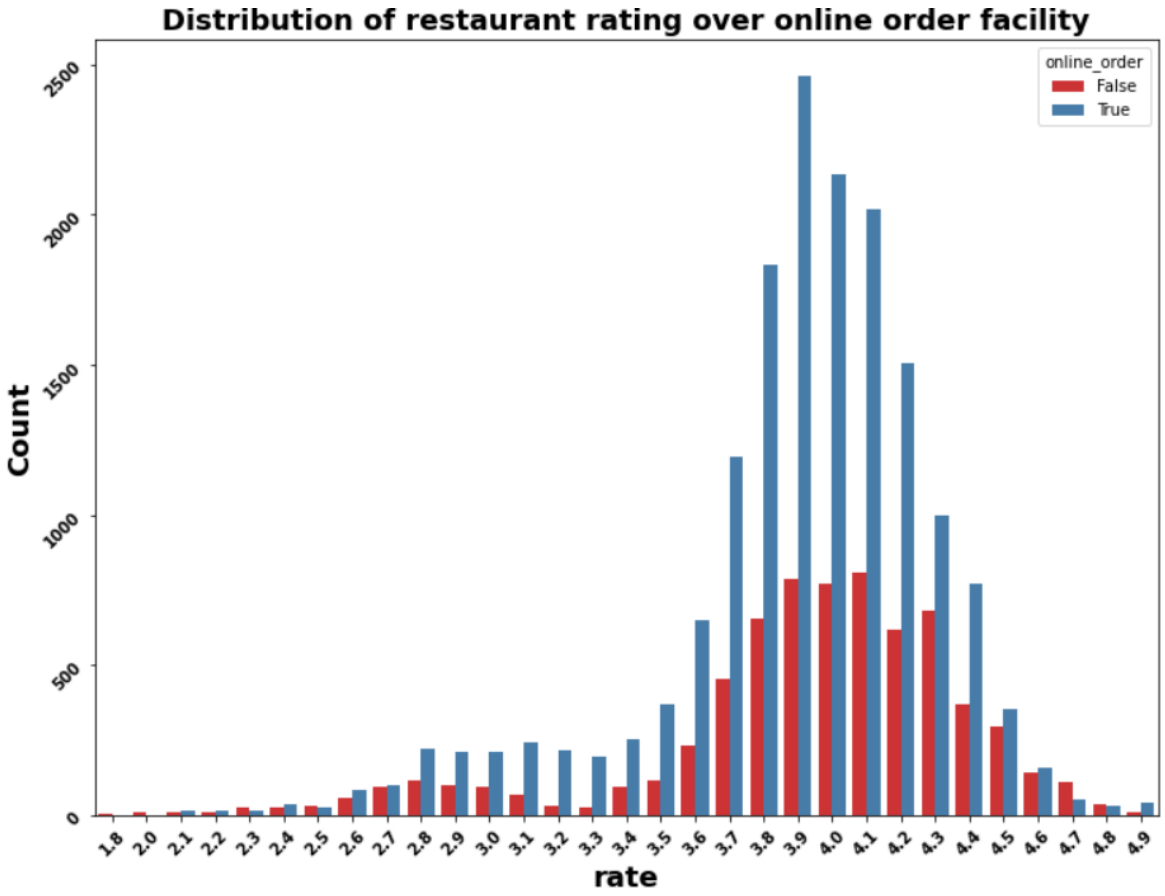
Let us see how many restaurants have the option for ordering online.

```
(array([    0.,  2000.,  4000.,  6000.,  8000., 10000., 12000., 14000.,
        16000., 18000.]), <a list of 10 Text major ticklabel objects>)
```



We see that majority of the restaurants(17,000+) have the option for booking online.

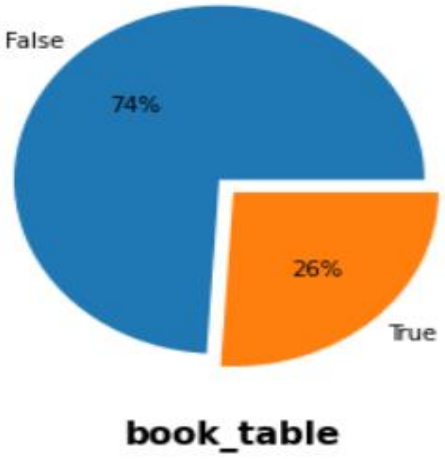
Now let us look at the number of restaurants for each rating where you can or can not book order online.



We see that the distribution for both yes or no values of *online_order* feature is somewhat same.

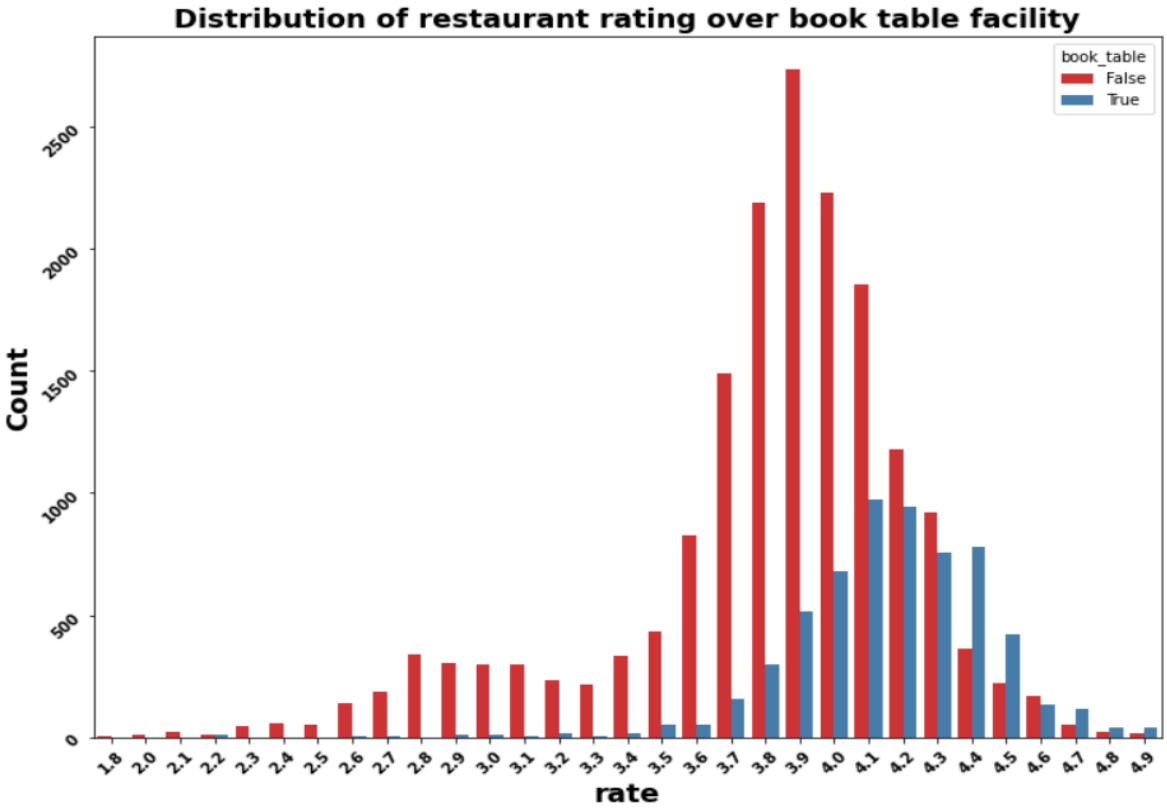
Majority of restaurants have rating 3.6 to 4.4, no matter whether you can order online or not.

Now let us see in how many restaurants you can book a table.



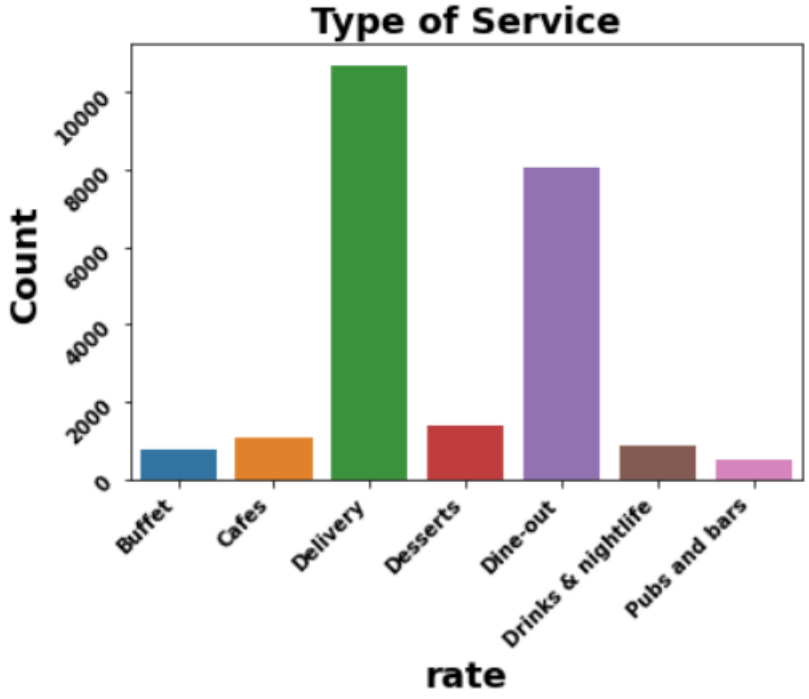
Almost 74 percent of restaurants in Banglore do not provide table booking facility.and 26 percent of restaurant provide table booking In India you cannot find table booking facility in any average restaurants,usually only five star restaurants provides table booking.

let us look at the number of restaurants for each rating as per the values(yes or no) for *book_table* feature



Now let us see, how many restaurants offer different types of meals.

```
FutureWarning
(array([    0.,  2000.,  4000.,  6000.,  8000., 10000., 12000.
 <a list of 7 Text major ticklabel objects>)
```

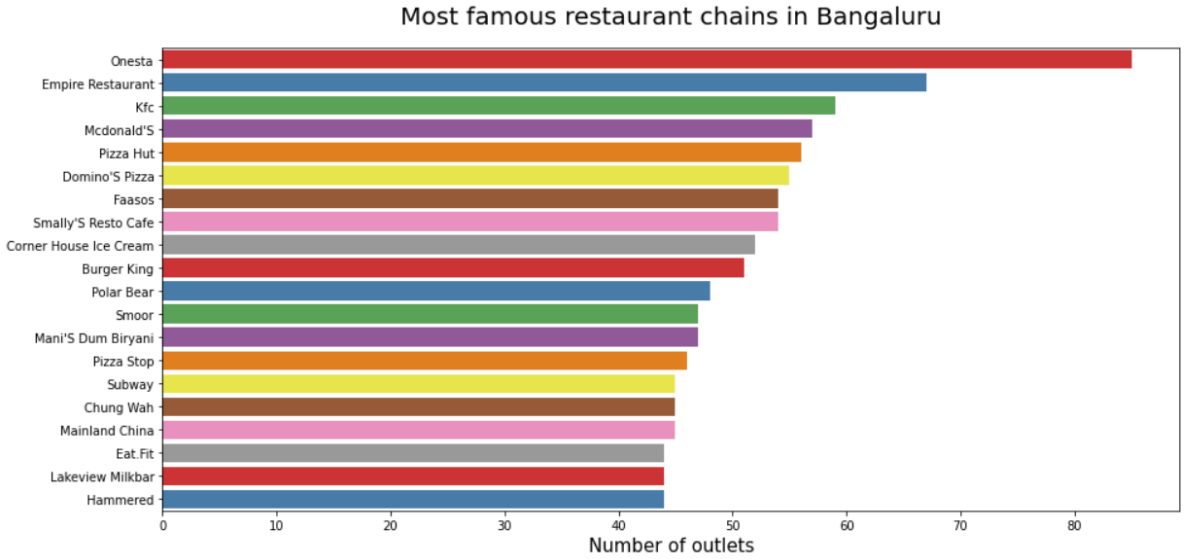


We see that the majority of restaurants offer delivery, which might account for the above conclusion that people at Bangalore prefer to eat at home.
Pubs and Bars and Buffet are offered by very less number of restaurants.

Most famous restaurant chains in Bengaluru.

```
[ ]  1 plt.figure(figsize=(15,7))
     2 chains=zomato['name'].value_counts()[:20]
     3 sns.barplot(x=chains,y=chains.index,palette='Set1')
     4 plt.title("Most famous restaurant chains in Bangaluru",size=20,pad=20)
     5 plt.xlabel("Number of outlets",size=15)
```

```
Text(0.5, 0, 'Number of outlets')
```



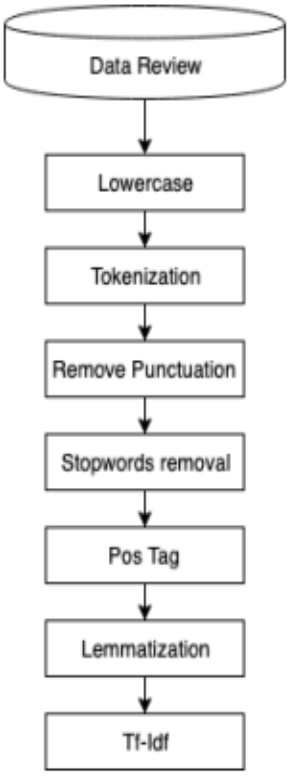
As you can see Cafe coffee day,Onesta,Just Bake has the most number of outlets in and around bangalore.

Chapter 4

4.1 Model Building

4.1.1 Algorithm Research and Selection:

It is well known for it's widely used in applications in recommender systems using Natural Language Processing (NLP) and K-Nearest Neighbors (KNN)



Flow Chart of NLP

NLP:

Natural Language Processing (NLP) is a field at the intersection of computer science, artificial intelligence, and linguistics. The goal is for computers to process or "understand" natural language in order to perform tasks like Language Translation and Question Answering.

With the rise of voice interfaces and chatbots, NLP is one of the most important technologies of the information age a crucial part of artificial intelligence. Fully understanding and representing the meaning of language is an extremely difficult goal.

Lowercase:

Lowercasing ALL your text data, although commonly overlooked, is one of the simplest and most effective form of text preprocessing. It is applicable to most text mining and NLP problems and can help in cases where your dataset is not very large and significantly helps with consistency of expected output.Lowercasing the two makes them identical, causing the classifier to lose important predictive features. While lowercasing *is generally* helpful, it may not be applicable for all tasks. While lowercasing should be standard practice, I've also had situations where preserving the capitalization was important.

Tokenization :

Tokenization is the process of splitting the given text into smaller pieces called tokens. Words, numbers, punctuation marks, and others can be considered as tokens.Tokenization is a common task in Natural Language Processing (NLP). It's a fundamental step in both traditional NLP methods like Count Vectorizer and Advanced Deep Learning-based architectures like Transformers.

Remove Pantuation :

The tokenizer can **replace all punctuation marks with themselves by adding a space around them.** Then it uses the space ("\S+") to split the text into tokens.

Remove Stopwords :

Removing stopwords is not a hard and fast rule in NLP. It depends upon the task that we are working on. For tasks like text classification, where the text is to be classified into different categories, stopwords are removed or excluded from the given text so that more focus can be given to those words which define the meaning of the text.

Lemmitization :

Lemmatization, on the other hand, is an organized & step-by-step procedure of obtaining the root form of the word. It makes use of vocabulary (dictionary importance of words) and morphological analysis (word structure and grammar relations).

TF-IDF:

TF-IDF which means **Term Frequency and Inverse Document Frequency,** is a scoring measure widely **used** in information retrieval (IR) or summarization. **TF-IDF** is intended to reflect how relevant a term is in a given document.

K-Nearest Neighbors (KNN):

KNN algorithm:- K-Nearest Neighbours In pattern recognition, the k-nearest neighbour's algorithm (k-NN) is a non-parametric method used for classification and regression. In both cases, the input 21 consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbour.

K-Nearest Neighbors (KNN) is one of the simplest algorithms used for regression and classification problems in Machine Learning. This takes the data and classifies new data points based on measures of similarity (e.g., distance function). Classification to its neighbour's is achieved by majority vote.

Random Forest:

The random forest is a classification algorithm that consists of several trees for decisions. While constructing each individual tree, it uses bagging and features variability to try to construct an uncorrelated forest of trees whose prediction by committee is more reliable than that of any individual tree

4.2 Algorithm Implementation :

1.Mount Google drive for loading dataset

2.Upload Image

3.Import all libraries like NumPy, Pandas, Matplotlib,

4.Reading dataset is proper or not

5.Perform Data Cleaning Steps for dataset

6.Visualize all data using visualization plots

7.Perform Text Processing on datasets

8.Build the model using TF-IDF and cosine similarity

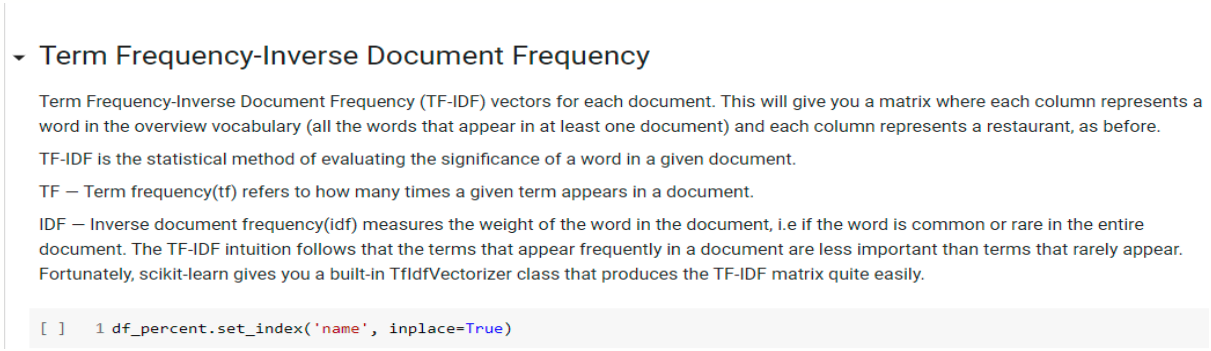
9.Apply recommendation function to recommend the restaurant name

10. Label Data for input like Training Data, Testing Data

11.perform train test split on our datasets

12.Apply KNN model based on cosine similarity

Using TF-IDF :



```
[ ]  1 recommend('Jalsa')
```

TOP 10 RESTAURANTS LIKE Jalsa WITH SIMILAR REVIEWS:

	cuisines	Mean Rating	cost
Truffles	Cafe, American, Burger, Steak	4.6	900.0
Jalsa Gold	North Indian, Mughlai, Italian	4.5	1.3
Nuts Over Salads Cafe	Cafe, Salad	4.4	800.0
Eat.Fit	Healthy Food, North Indian, Biryani	4.4	300.0
Toscano	Italian, Salad	4.4	1.5
Tree Tops Bar & Kitchen	North Indian, Chinese, Continental, Italian	4.4	1.0
Natural Ice Cream	Ice Cream, Beverages	4.3	200.0
Enerjuvate Studio & Cafe	Cafe, Continental, Healthy Food, Sandwich, Piz...	4.3	800.0
Easy Tiger	Finger Food, Continental	4.2	1.1
Manjit Da Dhaba	North Indian	4.2	300.0

Using KNN:

```
[ ]  1 recommendat('onesta')
```

```
Recommendations for smoor:

1: pizza hut, with distance of 1.0
2: rolls wheels, with distance of 1.0
3: onesta, with distance of 1.0
4: petoo, with distance of 1.0
5: kfc, with distance of 1.0
```

KNN

```
[69]  1 # K-Nearest Neighbor(KNN)
      2 from sklearn.neighbors import KNeighborsRegressor
      3 RegModel = KNeighborsRegressor(n_neighbors=3)
      4
      5 # Printing all the parameters of KNN
      6 print(RegModel)
      7
      8 # Creating the model on Training Data
      9 KNN=RegModel.fit(x_train,y_train)
     10 prediction=KNN.predict(x_test)
```

```
KNeighborsRegressor(algorithm='auto', leaf_size=30, metric='minkowski',
                    metric_params=None, n_jobs=None, n_neighbors=3, p=2,
                    weights='uniform')
```

```
[71]  1 # Custom Scoring MAPE calculation
      2 from sklearn.metrics import make_scorer
      3 custom_Scoring=make_scorer(Accuracy_Score, greater_is_better=True)
      4 # Importing cross validation function from sklearn
      5 from sklearn.model_selection import cross_val_score
      6
      7 # Running 10-Fold Cross validation on a given algorithm
      8 # Passing full data X and y because the K-fold will split the data and automatically choose train/test
      9 Accuracy_Values=cross_val_score(RegModel, x , y, cv=10, scoring=custom_Scoring)
     10 print('\nAccuracy values for 10-fold Cross Validation:\n',Accuracy_Values)
     11 print('\nFinal Average Accuracy of the model:', round(Accuracy_Values.mean(),2))
```

```
Accuracy values for 10-fold Cross Validation:
 [94.31613975 96.68909366 95.88733056 95.67367179 94.21324828 98.56422963
 98.69341916 96.52146446 94.99675554 94.07672244]

Final Average Accuracy of the model: 95.96
```

Chapter 5

5.1 Future Scope :

As the dataset is pretty straight forward we did not split the data for the future research can be continued to enhance the delivery option available making users to pick as there is no necessity for them to come to the outlets the best cuisines will be delivered to their locations and also in our research we observed direct walk in for table booking outlets and low taking reservation there is scope to work even in this part more by running of different models for better accuracy. For this we need more there is also possibility to add more features to know about the opinion of customers as why they are not preferring the delivery system, not ordering online, even increase the votes and so on. There is also chance to add versions to the models and follow approach production network. We can also train the algorithms used again for better analysis of features in static and real time. There is also scoped to retrain the paradigm for the studied model and can also find a pathway to check the paradigm production. To enable card payment. To add kitchen raw material management to the existing system. To extend the existing system to a chain of multiple restaurants.

Chapter 6

6.1 Conclusion

Our system implements recommendation system to provide restaurant recommendations based on their previous reviews and rating.The recommendations are pretty much accurate as per our tests. Our system can be easily extended to other cities and cuisines. Our system has immense potential and is multipurpose as it can come handy for businesses as well as the average user. The field of restaurant recommendations is one of the uncharted territories and our system is a small step in a giant ocean.a recommender system which suggests restaurants according to user's food item preference based on the reviews received by Zomato about these restaurants. We also kept in mind the side items favored while making the recommendation. We believe that this information may be useful for the extensive review data available on such sites.

Chapter 7

7.1 Reference

1. https://machinelearningmastery.com/clean-text-machine-learning-python/

2. https://thecleverprogrammer.com/2020/11/18/restaurant-recommendation-system-with-python/

3. https://thecleverprogrammer.com/2020/11/18/restaurant-recommendation-system-with-python/

4. https://www.slideshare.net/ShoaibKhan1994/test1-63741146?from_action=save

5. https://www.ipinnovative.com/journal-article-file/11195

6. https://ijircst.org/DOC/6_what-to-taste-a-food-recommendation-system.pdf

7. https://blog.dominodatalab.com/recommender-systems-collaborative-filtering/