Analysis of TripAdvisor Hotels' Customer reviews using Sentiment Analysis

BIA 660-B: Web Mining

Instructor: Jingyi Sun

Group 4

Ronak Kachalia (10459133)

Devanshi Mehta (10459130)

Krina Shah (10459200)

Sacheth Shetty (10459184)

Introduction

Analysis of user comments and reviews can help businesses in understanding how their customers are feeling about their products and services, which in turn provides deep insights to major stakeholders in the business on how to improve specific areas of products and services.

TripAdvisor is a travel company that assists its customers in finding the best rates for their hotel stay as well as booking tickets for their trip. One of the services it offers is their comprehensive hotel booking suite which enables its users to not only view hotels based on location, cost, cleanliness, and various other factors but also review the stay of other travelers at those hotels. The users are prompted to write a text-based review of more than 200 characters and provide an overall rating as well as a rating for cleanliness, rooms, and location as part of their review. Users can read thousands of reviews left by other users for a specific hotel before making their choice. These reviews are not only useful for other users, but they provide several insights to major stakeholders for the hotels which might help them improve the quality of their services.

TripAdvisor sticks to three main ratings for a specific hotel, namely cleanliness, rooms and location of the hotel, along with an overall rating for the hotel. However, it is not necessary that the guests are always looking for these specific services in the hotel. Adversely, the review left by the user might include more details about services which they might be unhappy about, however the overall numerical rating does not provide any information regarding the details of those services.

For example, A guest might be satisfied with the cleanliness of the hotel, their room size as well as the location, but they might be extremely unhappy with other services such as food or value for money. The guests might express these concerns in their text review and change the overall rating for the hotel, but this numerical rating does not provide enough information to the Hotel's management team to make changes or improve their services.

Our project aims at bridging the gap between these text-based reviews using Sentimental Analysis as well as identifying certain other categories from popular words used in the review text which users have left for specific hotels. These new categories not only help the user's narrow down their search for their perfect stay, but also helps the businesses to ascertain which services need to be improved in order to increase customer satisfaction, and bring in more business into their respective hotels.

Literature Review

We reviewed the work detailed in the paper by Hsiu-Yuan Tsao and Ming-Yi Chen "The asymmetric effect of review valence on numerical rating", where the authors have conducted a sentiment analysis via text mining, using self-developed computer programs to retrieve a data set from the TripAdvisor website. This study finds there is an asymmetric relationship between review valence or the verbal review text and numerical rating. The authors further find brand strength to have an important moderating role. For a stronger brand, negative review content will have a greater impact on numerical ratings than positive review content, while for a weaker brand, positive review content will have a greater impact on numerical ratings than negative review content.

Therefore, the overall rating that is provided to a hotel is not a reliable measure of services offered by a specific hotel branch or customer satisfaction. The authors mention that assumption verbal review text is symmetrically related to the numerical rating might be a false one, since brand image is a significant factor that customers consider while writing these reviews on TripAdvisor. Similarly, other factors or services offered by a specific hotel might not be considered while providing their independent overall rating to the hotel. The authors further conclude that marketers could adopt sentiment analysis via text mining of online reviews as a valid measure or predictor of consumer satisfaction or numerical ratings. Strong brands should direct more attention to negative reviews, because in such reviews the negative impact transcends the positive. In contrast, weak brands should aim to exploit as many positive reviews as possible to minimize the impact of any negative reviews.

We noted that part the "Brand Image" of the Hotel is simply just one of the factors that might affect the Review valence and overall rating. Other factors would include the services offered by the specific Hotel Branch, such as the quality of food and dining services, gym and fitness services, staff politeness, etc. All of these keywords can be identified and a sentiment analysis would provide us with more insights as to whether the customer reviewing the hotel had a positive or negative experience on these specific factors. This might in turn help us to bridge the gap between the review valence and the overall rating provided by TripAdvisor.

Research Area

Analyzing customer reviews for:

1. extracting categories of services of hotels like room, gym, value for money, etc.,
2. performing sentimental analysis on textual information regarding each category to classify individual extracted services as positive or negative review
3. calculating new rating by taking weights of individual categories of services in the reviews

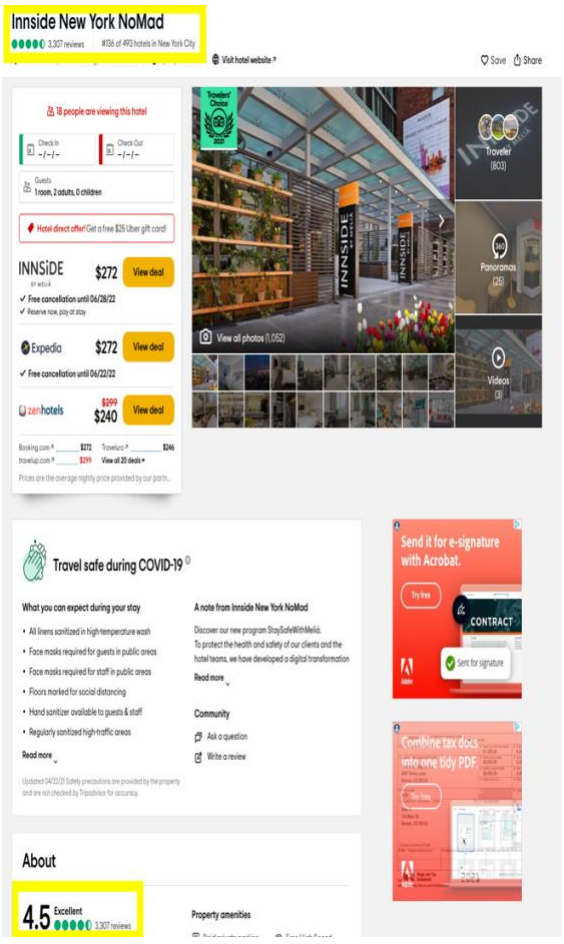
Objective

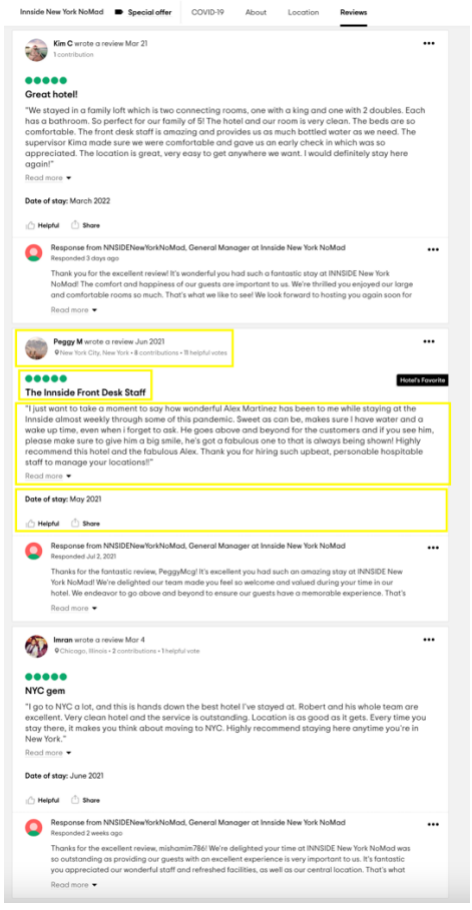
Based on the research question imposed, this experiment will involve various steps before we actually work on building models for review analysis.
In this section, we will focus on extracting data from TripAdvisor, pre-processing the extracted data, performing Exploratory Data Analysis, and drawing insightful conclusions.

Implementation

A. Data Extraction

- In order to extract data from TripAdvisor, we will be implementing web scraping using Selenium.
- The bracket for number of hotels to scan is restricted to 6 hotels, and for each hotel we will be scraping 20 review pages where each page constitutes of 10 different reviews.





- The above pictures of the TripAdvisor screens illustrate the fields we are scraping using. Selenium. Below are the fields we will be concerned with:
    - Name of the hotel
    - Overall ratings
    - Number of reviews
    - Username of reviewer
    - Review date
    - No. of contributions
    - No. of votes review received
    - Reviewer's overall ratings
    - Review title
    - Review text
    - Date of stay
    - Individual category ratings (if any)

- Below are the screenshots of the Web Scraping scripts implemented using Selenium in Python:

In [1]:
```
import csv
from selenium import webdriver
import time
from selenium.webdriver.common.by import By
```

In [2]:
```
urls = [
    "https://www.tripadvisor.com/Hotel_Review-g60763-d8873263-Reviews-Innside_New_York_NoMad-New_York_City_New_York.htm
    "https://www.tripadvisor.com/Hotel_Review-g60763-d23448880-Reviews-Motto_by_Hilton_New_York_City_Chelsea-New_York_C
    "https://www.tripadvisor.com/Hotel_Review-g60763-d12551350-Reviews-Pod_Times_Square-New_York_City_New_York.html",
    "https://www.tripadvisor.com/Hotel_Review-g60763-d7182804-Reviews-Residence_Inn_New_York_Downtown_Manhattan_world_T
    "https://www.tripadvisor.com/Hotel_Review-g60763-d5040757-Reviews-Homewood_Suites_by_Hilton_New_York_Midtown_Manhat
    "https://www.tripadvisor.com/Hotel_Review-g60763-d8515751-Reviews-Hotel_Riu_Plaza_New_York_Times_Square-New_York_Ci
]
```

In [6]:
```
# open the file to save the review
path_to_file = '/Users/ronak/Desktop/HotelReviews.csv'
csvFile = open(path_to_file, 'a', encoding="utf-8")
csvWriter = csv.writer(csvFile)
csvWriter.writerow(['link', 'hotel_name', 'overall_rating',  'date_of_review', 'user_name',
                    'review_rating', 'review_title', 'review_text', 'date_of_stay', 'contributions',
                    'helpful_votes', 'value_rating', 'rooms_rating', 'location_rating', 'clean_rating',
                    'service_rating', 'sleep_rating'])
```

In [9]:
```
for link in urls:
    path_to_file = link
    driver = webdriver.Safari()
    driver.get(link)
    time.sleep(2)

    header = hotel_name = ''
    hotel = None
    overall_rating = 0

    # header of hotel
    try:
        header = driver.find_element(By.XPATH, ".//div[contains(@data-test-target, 'hr-aft-info')]")
    except:
        print('Error: header of hotel')

    # hotel name
    try:
        hotel = header.find_elements(By.XPATH, ".//div[contains(@class, 'eIsCM f')]")
        hotel_name = hotel[0].text
    except:
        print('Error: hotel name')

    # overall rating
    try:
        overall_rating = header.find_element(By.XPATH, ".//span[contains(@class, 'ui_bubble_rating bubble_')]").get_att
        overall_rating = int(overall_rating)/10
    except:
        print('Error: overall rating')

    pages_to_scrape = 20

    for i in range(0, pages_to_scrape):
        print('Page no.: ', i)

        time.sleep(2)

    #     driver.find_element(By.XPATH, ".//div[contains(@data-test-target, 'expand-review')]").click()

        container = driver.find_elements(By.XPATH, "//div[@data-reviewid]")
        user_header_1 = driver.find_elements(By.CLASS_NAME, 'bcaHz')
        user_header_2 = driver.find_elements(By.CLASS_NAME, 'BZmsN')

        for j in range(len(container)):
            print('Review no.: ', j)

            date_of_review = user = location = title = review = date_of_stay = ''
            read_more = None
            contributions = helpful_votes = rating = 0
            value_rating = rooms_rating = location_rating = clean_rating = service_rating = sleep_rating = 0
            # header of the review

            # review date
            try:
                date_of_review = " ".join(user_header_1[j].text.split(" ")[-2:])
            except:
                print('Error: review date')

            # username
            try:
                user = user_header_1[j].text.split("wrote")[0]
            except:
                print('Error: user')

            # location of user
            try:
                location = user_header_2[j].find_element(By.XPATH, ".//span[contains(@class, 'default ShLyt small')]").
            except:
                print('Error: location')

            # contributions and votes
            try:
                contr_and_votes = user_header_2[j].find_elements(By.XPATH, ".//span[contains(@class, 'eUTJT')]")
                for k in contr_and_votes:
                    k = k.text.split(' ')
                    if k[1] == 'contributions':
                        contributions = k[0]
                    if k[1] == 'helpful':
                        helpful_votes = k[0]
            except:
                print('Error: contributions and votes')

            # main section

            # click on read more of the reviews
            try:
                read_more = container[j].find_element(By.XPATH, ".//div[contains(@data-test-target, 'expand-review')]")
                #driver.execute_script("arguments[0].click();", read_more)
            except:
                print('Error: read more')

            # ratings
            try:
                rating = container[j].find_element(By.XPATH, ".//span[contains(@class, 'ui_bubble_rating bubble_')]").g
                rating = int(rating)/10
            except:
                print('Error: ratings')
```

```
            # title of review
            try:
                title = container[j].find_element(By.XPATH, ".//div[contains(@data-test-target, 'review-title')]").text
            except:
                print('Error: title')

            # review text
            try:
                review = container[j].find_element(By.XPATH, ".//q[@class='XllAv H4 _a']").text.replace("\n", "  ")
            except:
                print('Error: review text')

            # date of stay
            try:
                date_of_stay = container[j].find_element(By.XPATH, ".//span[contains(@class, 'euPKI _R Me S4 H3')]").te
                date_of_stay_child = container[j].find_element(By.XPATH, ".//span[contains(@class, 'CrxzX')]").text
                date_of_stay = date_of_stay.replace(date_of_stay_child, '')
            except:
                print('Error: date of stay')

            # Additional ratings
            try:
                additional_ratings = container[j].find_elements(By.XPATH, ".//div[contains(@class, 'fFwef S2 H2 cUidx')
                for k in additional_ratings:
                    x = k.find_element(By.XPATH, ".//span[contains(@class, 'ui_bubble_rating bubble_')]").get_attribute
                    x = int(x)/10
                    if k.text == 'Value':
                        value_rating = x
                    elif k.text == 'Rooms':
                        rooms_rating = x
                    elif k.text == 'Location':
                        location_rating = x
                    elif k.text == 'Cleanliness':
                        clean_rating = x
                    elif k.text == 'Service':
                        service_rating = x
                    elif k.text == 'Sleep Quality':
                        sleep_rating = x
                    else:
                        pass
            except:
                print('Error: additional ratings')

            # write to CSV
            try:
                csvWriter.writerow([link, hotel_name, overall_rating,  date_of_review, user, rating, title, review, dat
            except:
                print('Error: writing data to CSV')

        # Next page
        try:
            driver.find_element(By.XPATH, './/a[@class="ui_button nav next primary "]').click()
        except:
            print('Error: next page')
            break

    driver.quit()
```

- Scrapped Data in CSV:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	link	hotel_name	overall_ratin	date_of_revi	user_name	review_ratin	review_title	review_text	date_of_stay	contributions	helpful_vote	value_rating	rooms_ratin	location_rati	clean_rating	service_ratin	sleep_rating	
2	https://www	Innside New	45	review Yeste	Kim C	50	Great hotel!	We stayed ir	March 2022	0	0	0	0	0	0	0	0	
3	https://www	Innside New	45	21-Jun	Peggy M	50	The Innside f	I just want t	May 2021	8	11	0	40	0	50	50	0	
4	https://www	Innside New	45	4-Mar	Imran	50	NYC gem	I go to NYC a	June 2021	2	1	50	0	50	0	50	0	
5	https://www	Innside New	45	2-Mar	Jay B	50	Great Experi	I don't usuall	February 20	0	1	50	50	0	0	50	0	
6	https://www	Innside New	45	22-Feb	Jeweliana15	50	Amazing Hos	I've stayed h	February 20	25	4	50	0	0	0	50	0	
7	https://www	Innside New	45	22-Feb	Jonathan Pill	50	FANTASTIC T	I am a native	February 20	0	0	0	0	0	50	50	50	
8	https://www	Innside New	45	22-Feb	EJL	20	Way better c	I was so exci	February 20	0	0	0	0	0	0	0	0	
9	https://www	Innside New	45	22-Jan	Jay C	50	Great Weeke	Spent a Satu	January 202	23	11	0	0	50	50	50	0	
10	https://www	Innside New	45	22-Jan	gravadorrom	50	one of the be	staff are sup	January 202	2	0	50	50	50	50	50	50	
11	https://www	Innside New	45	22-Jan	Stacey D	40	Great locatic	This hotel wa	January 202	0	0	50	40	50	40	40	30	
12	https://www	Innside New	45	22-Jan	Matthew W	40	Good week,	Good week i	January 202	3	0	0	0	0	0	0	0	
13	https://www	Innside New	45	22-Jan	Mark R	40	Ideally situat	We stayed a	January 202	303	107	0	0	0	0	0	0	
14	https://www	Innside New	45	22-Jan	nikkilogancui	50	Perfect Stay,	My husband	January 202	0	0	0	0	0	0	0	0	
15	https://www	Innside New	45	22-Jan	etremat	50	NYE in New	Stayed at thi	December 2	60	50	0	0	0	0	0	0	
16	https://www	Innside New	45	21-Dec	Maria R	50	Excellent Hot	Is was a wor	December 2	0	0	0	0	0	0	0	0	
17	https://www	Innside New	45	21-Dec	KyleT_Bethe	50	XMas visit to	Location in tl	December 2	2	0	0	0	0	0	0	0	
18	https://www	Innside New	45	21-Dec	Kimbers69	40	Surprise Chri	On arrival we	December 2	2	0	0	0	0	0	0	0	
19	https://www	Innside New	45	21-Dec	AMK-flyer	20	Shocking exp	A shocking e	December 2	0	1	0	0	0	0	0	0	
20	https://www	Innside New	45	21-Dec	Declan S	50	Perfect Spot	Great hotel,	December 2	3	8	0	0	0	0	0	0	
21	https://www	Innside New	45	21-Dec	NRush45	30	Good Locatic	The hotel wa	December 2	4	1	0	0	0	0	0	0	
22	https://www	Innside New	45	21-Dec	MHS312	50	Great Hotel -	The hotel wa	December 2	8	8	0	0	0	0	0	0	
23	https://www	Innside New	45	21-Dec	Ana	20	Disappointec	Bathroom dc	December 2	8	2	0	0	0	0	0	0	
24	https://www	Innside New	45	21-Dec	micheleseftc	40	Great hotel i	Location of t	December 2	16	12	0	0	0	0	0	0	
25	https://www	Innside New	45	21-Dec	karenHA27	50	Nice hotel ar	We stayed ir	December 2	0	0	0	0	0	0	0	0	
26	https://www	Innside New	45	21-Dec	CMC2008	30	Great hotel,	The rooms fc	December 2	22	9	0	0	0	0	0	0	
27	https://www	Innside New	45	21-Dec	cthomas019	50	Great Place	Glad I payed	December 2	0	0	0	0	0	0	0	0	
28	https://www	Innside New	45	21-Dec	T L	40	Great hotel,	I stay at this	December 2	48	15	0	0	0	0	0	0	
29	https://www	Innside New	45	21-Dec	Fil G	50	Great NYC h	Staff are sup	December 2	7	8	0	0	0	0	0	0	
30	https://www	Innside New	45	21-Dec	AW	30	Good locatio	Hotel and roo	December 2	3	0	0	0	0	0	0	0	
31	https://www	Innside New	45	21-Dec	megirod1	50	Great quiet l	Nice and qui	November 2	0	0	0	0	0	0	0	0	
32	https://www	Innside New	45	21-Dec	Maria M	10	Stay away fr	The mattress	November 2	0	1	0	0	0	0	0	0	
33	https://www	Innside New	45	21-Dec	mvinci	50	Welcome ho	This is my gc	November 2	21	12	0	0	0	0	0	0	
34	https://www	Innside New	45	21-Nov	Elizabeth C	40	Amazing Fro	I can't thank	November 2	3	1	0	0	0	0	0	0	
35	https://www	Innside New	45	21-Nov	kylienm12	40	Very chic, we	The hotel wa	November 2	0	0	0	0	0	0	0	0	
36	https://www	Innside New	45	21-Nov	Stevi G	40	Nice, modern	We had a do	November 2	130	77	0	0	0	0	0	0	
37	https://www	Innside New	45	21-Nov	RozUsa	40	Great stay!	Hotel is cent	November 2	6	0	0	0	0	0	0	0	
38	https://www	Innside New	45	21-Nov	Alejandra M	50	Great Stay	I visited NY (	November 2	2	1	0	0	0	0	0	0	
39	https://www	Innside New	45	21-Nov	kaleighcarola	50	Friendly Staf	Our stay was	November 2	0	0	0	0	0	0	0	0	

B. Date Pre-processing

The scraped data from the TripAdvisor website included a number of columns where the data provided was inconsistent. Many columns in the scraped data contained both an object of strings, single string values as well as integer values in the same column. Therefore, we used certain pre-processing steps in order to clean the data and make it a little more consistent in order to perform exploratory data analysis. This will provide us with useful insights into the scrapped hotel review data.

- **Dropped column 'link' which is not useful for our analysis**

  The 'link' column contained a URL link which redirected to the actual review in the TripAdvisor website. We concluded that this data was not useful to us in order to perform any analysis.

```
#dropping the review lnk column since we do not require that for analysis
df.drop('link', axis=1, inplace=True)
df.head()
df.dtypes
```

Figure 1: Script for dropping 'link' column

- **Replaced all values in the 'date_of_review' column which contained string values such as 'reviewed today' or 'reviewed yesterday' with consistent date value.**
  The column 'date_of_review' contained values such as 'reviewed today' & 'reviewed yesterday' which was inconsistent with the other values in the column which were in the format of a data value. (21-Mar or YY- shorthand month name). We replaced these values with the current month and year value of '22-Mar'.

```
#In column date_of_review, replacing certain string values to a date value
#For e.g replaced "review Yesterday" with 22-March (YY/Month)

df['date_of_review'] = df['date_of_review'].replace(['review Yesterday'],'22-Mar')
df['date_of_review'] = df['date_of_review'].replace(['review Today'],'22-Mar')
df
```

Figure 2: Replacing String values in date columns

- **We noted that the review rating columns had review scores out of a 50. We reduced these scores by a factor of 10 in order to make the review ratings simple.**
  We noted that the columns containing integer review ratings for the below given rating columns, contained an integer score out of a 50 which we reduced by a factor of 10 in order to keep it consistent with the reviews present in the Trivago website.
  1. Overall rating
  2. Value rating
  3. Location rating
  4. Clean Rating

5. Service Rating
6. Sleep Rating

```
#reducing ratings by a factor of 10
df['review_rating']=df['review_rating']/10
df['value_rating']=df['value_rating']/10
df['rooms_rating']=df['rooms_rating']/10
df['location_rating']=df['location_rating']/10
df['clean_rating']=df['clean_rating']/10
df['service_rating']=df['service_rating']/10
df['sleep_rating']=df['sleep_rating']/10
df.head()
```

Figure 3: Reducing the factor of review score by 10

- **We noted that columns of 'date_of_stay' and 'date_of_review' contained different date formats.**
  We noted that the above two columns contained values in 2 different date formats.
  In order to perform our EDA over the months in a year, we extracted only the month parameter in the date values from both the columns and stored them in new columns 'month_of_review' and 'month_of_stay'.

```
#Extracting only the months from date_of_review and date_of_stay

df['month_of_review']= df['date_of_review'].str.split('-',expand=True)[1]
df['month_of_stay']=df['date_of_stay'].str.split(' ',expand=True)[1]
df.head()
```

Figure 4: Extracting only the month from the date columns

- **Based on the months extracted from the 'date_of_review' & 'date_of_stay' column values, we grouped the reviews based on the quarter of the year.**
  We also clubbed the reviews based on the financial quarters in a year (Q1,Q2,Q3,Q4) based on the values in the above columns and stored these values in 2 different columns, 'quarter_of_review' & 'quarter_of_stay'. We will further use these new columns to perform EDA in order to understand if there are any insights we can gain based on quarter wise review distribution.

```
#Assigning the month_of_review to quarters in a year
#del df['Quarter']
q1=['Jan','Feb','Mar']
q2=['Apr','May','Jun']
q3=['Jul','Aug','Sep']
q4=['Oct','Nov','Dec']
df.loc[df['month_of_review'].str.contains('|'.join(q1)), 'quarter_of_review'] = 'Q1'
df.loc[df['month_of_review'].str.contains('|'.join(q2)), 'quarter_of_review'] = 'Q2'
df.loc[df['month_of_review'].str.contains('|'.join(q3)), 'quarter_of_review'] = 'Q3'
df.loc[df['month_of_review'].str.contains('|'.join(q4)), 'quarter_of_review'] = 'Q4'
df.head()
```

Figure 5: Making a new column for quarter of review

```
#Assigning the month_of_stay to quarters in a year
q1=['January','February','March']
q2=['April','May','June']
q3=['July','August','September']
q4=['October','November','December']
df.loc[df['month_of_stay'].str.contains('|'.join(q1)), 'quarter_of_stay'] = 'Q1'
df.loc[df['month_of_stay'].str.contains('|'.join(q2)), 'quarter_of_stay'] = 'Q2'
df.loc[df['month_of_stay'].str.contains('|'.join(q3)), 'quarter_of_stay'] = 'Q3'
df.loc[df['month_of_stay'].str.contains('|'.join(q4)), 'quarter_of_stay'] = 'Q4'
df.head()
```

Figure 6: Making a new column for quarter of stay

C. Exploratory Data Analysis (EDA)

1. Correlation

Initially to get a better understanding about dependent features in our data set we plot the correlation between our variables using heat map.

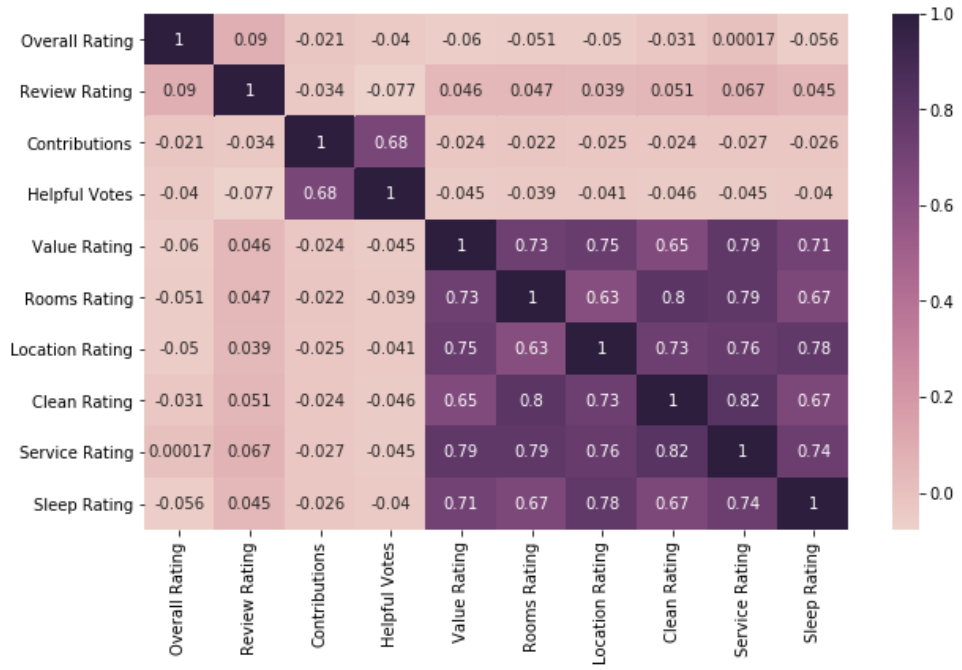


Figure 7: Correlation matrix

Here, we can observe that Ratings are highly correlated.

2. Hotel Name v/s Review rating

We plotted a bar graph that displays the count of unique review ratings per hotel.

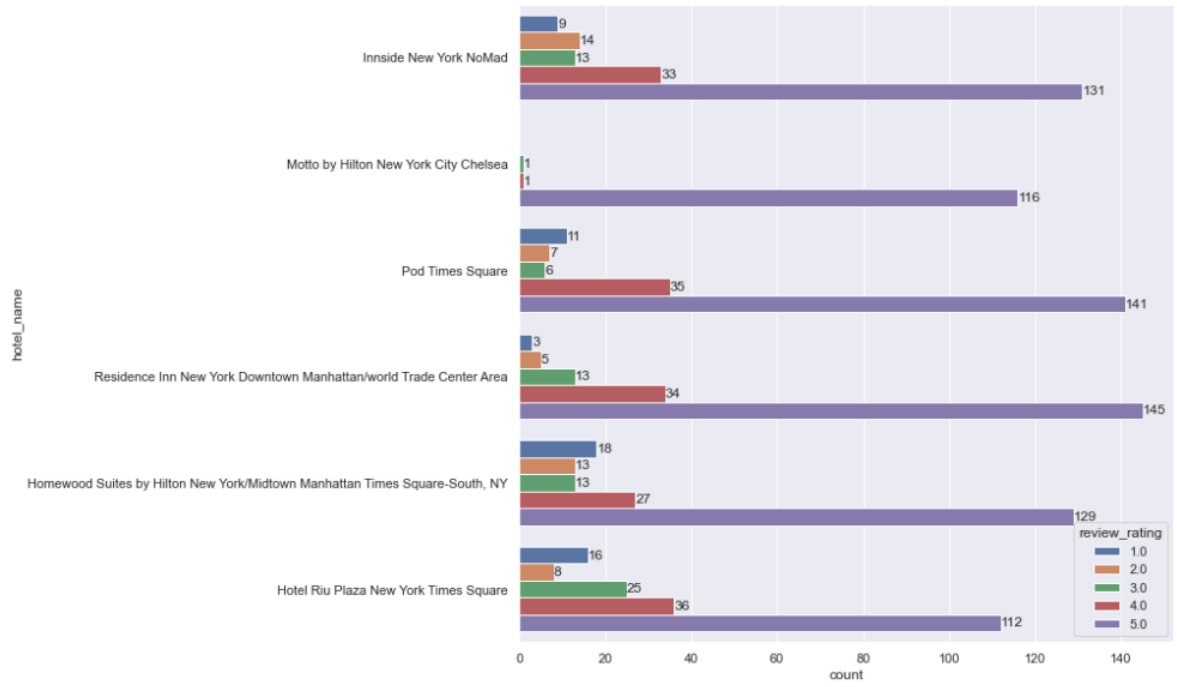


Figure 8: Plot of Hotel_name v/s Review Rating

Here, we can observe that most of the hotels have a high density of 5 rating. This means that the majority of users have reviewed most of the Hotel and provided them with a rating of 5. Also, hotels like 'Motto by Hilton New York Chelsea' do not have any reviews which have 1 or 2 stars rating.

3. **Users who have provided a review 5 times before(contribution>5)**
   Here we will say that assuming that the reviewers who have provided a review on TripAdvisor before give better reviews in terms of the "review_text" quality. Therefore we specifically targeted users who have provided more than 5 reviews.

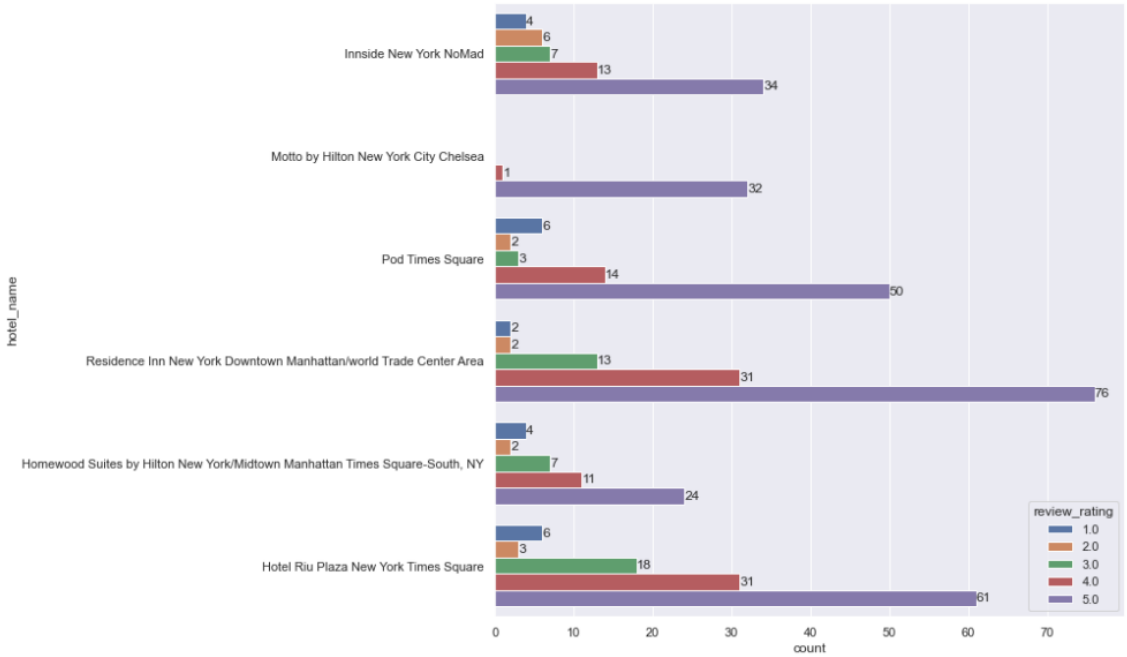


Figure 9: Plot of Hotel_name v/s Review Rating for user who have provided 5 or more reviews

4. Review Rating for a hotel every quarter

We plotted a graph in order to visualize the number of reviews for each hotel in our dataset based on the quarter in which these reviews were provided on the TripAdvisor website. We observed based on our research on the Hospitality industry that as part of the Hotel's business model, there are changes made to the hotel every season and every quarter in order to keep their amenities new and fresh. At the same time, certain amenities or services may not be offered by the Hotel all year round, for e.g., an "Outdoor Pool and Bar" might not be available in the hotel during the winter months.

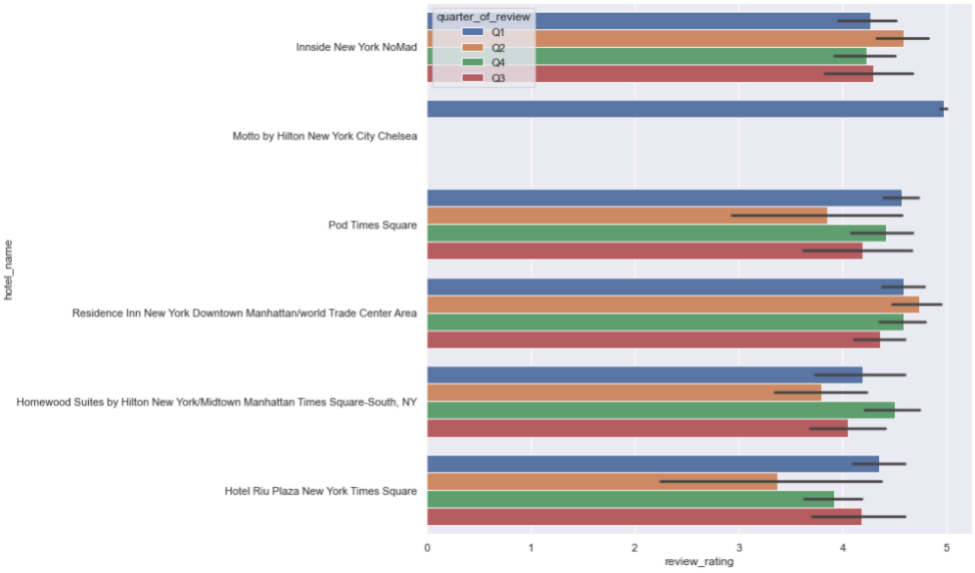


Figure 10: Plot of containing count of review rating per hotel for every quarter

We noted that for most hotels, we had a more or less even distribution of reviews which were provided during each quarter. We further noted that for the hotel "Motto by Hilton New York City Chelsea", most of the reviews were provided during Q1. Upon further research we noted this hotel was inaugurated in January 2022, which is why most of the reviews are provided in Q1

5. **Word cloud of review title**

Word Cloud displays the most prominent or frequent words in a body of text. Here we display the word cloud of review title in our dataset. We try to find the most frequent words used in the "review title" column.

Figure 11: Word cloud of review with all different ratings

As seen in barplot "Hotel Name Vs Review rating" we have a high density of 5 rating. Thus, We have a high frequency of positive words in review titles like excellent, great. We plotted word cloud of review title where review rating is 1,2,3,4,5 individually to look at the frequent words used in their respective review rating and try to establish correlation between the value given to review rating and value given to room rating, location rating, clean rating, service rating and sleep rating.

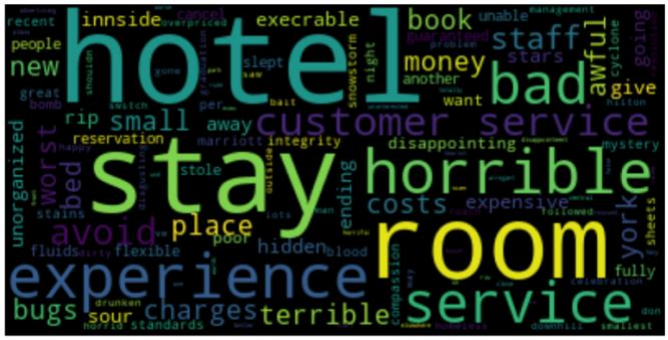


Figure 12: Word cloud of review title where rating = 1

Figure 13: Word cloud of review title where rating = 2

Figure 14: Word cloud of review title where rating = 3

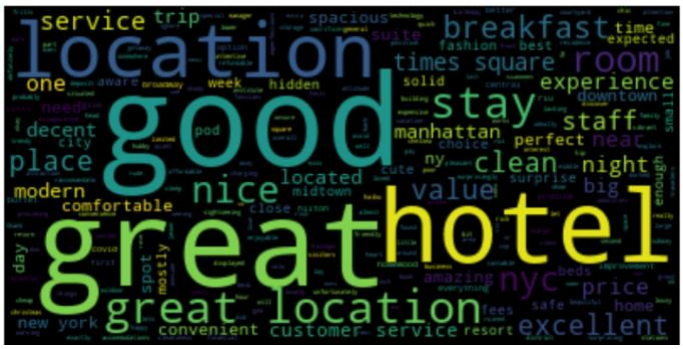


Figure 15: Word cloud of review title where rating = 4

Figure 16: Word cloud of review title where rating = 5

We got an insight from various word clouds from rating 1 to 5 that when rating was 1 the review title had negative words like horrible, bad. While moving forward toward rating 3

we observe a positive shift in the sentiment with words like better, nice. And when the rating was 5 we observe positive words like excellent, great, perfect.

We further obtained the world cloud for each rating based on hotel "Innside New York NoMad".

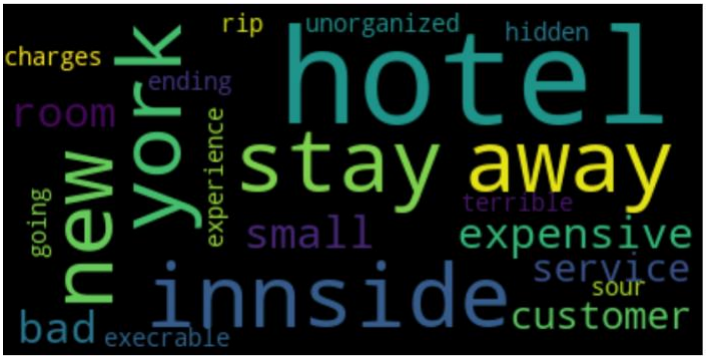


Figure 17: Word cloud of review title where rating = 1 for hotel "Innside New York NoMad"

Figure 18: Word cloud of review title where rating = 2 for hotel "Innside New York NoMad"

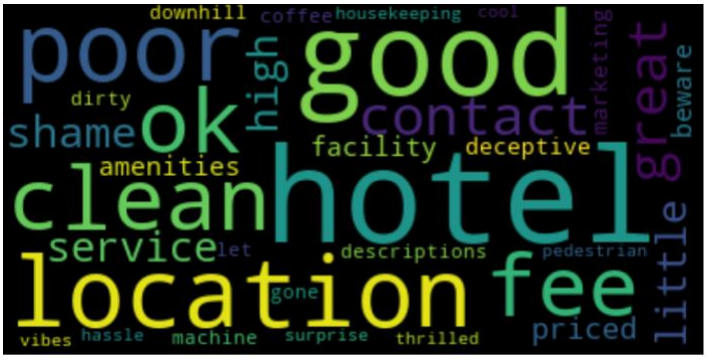


Figure 19: Word cloud of review title where rating = 3 for hotel "Innside New York NoMad"

Figure 20: Word cloud of review title where rating = 4 for hotel "Innside New York NoMad"

Figure 21: Word cloud of review title where rating = 5 for hotel "Innside New York NoMad"

Once we had obtained the word clouds for each and every rating present in the dataset, we also found the word clouds for each and every rating and review provided to each hotel on our dataset. We then compared the word clouds obtained in figures 11 through 15 with the word clouds we obtained below and noted that we did not see a significant difference in the results based on “hotel_name”.

Project Log

Tasks	Team Member(s)
Scraping data from TripAdvisor	Ronak Kachalia
Data cleaning and preprocessing	Sacheth Shetty, Krina Shah
Exploratory Data Analysis	Sacheth Shetty, Krina Shah
Report preparation	Ronak Kachalia, Sacheth Shetty, Krina Shah, Devanshi Mehta

References

- **The asymmetric effect of review valence on numerical rating:**
  Tsao, Hsiu-Yuan & Chen, Ming-Yi & Lin, Hao-Chiang & Ma, Yu-Chun (2018)

- **Apply word vectors for sentiment analysis of APP reviews**
  Xian Fan; Xiaoge Li; Feihong Du; Xin Li; Mian Wei (2016)
  3rd International Conference on Systems and Informatics (ICSAI)
  Year: 2016 | Conference Paper | Publisher: IEEE

- **Analyzing Sentiments Expressed on Twitter by UK Energy Company Consumers**
  Victoria Ikoro; Maria Sharmina; Khaleel Malik; Riza Batista-Navarro (2018)
  Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS)
  Year: 2018 | Conference Paper | Publisher: IEEE

- **Comparative study of Twitter Sentiment On COVID - 19 Tweets**
  Anu J Nair; Veena G; Aadithya Vinayak (2021)
  5th International Conference on Computing Methodologies and Communication (ICCMC)
  Year: 2021 | Conference Paper | Publisher: IEEE

- **A framework for sentiment analysis with opinion mining of hotel reviews**
  Kudakwashe Zvarevashe; Oludayo O. Olugbara (2018)
  Conference on Information Communications Technology and Society (ICTAS)
  Year: 2018 | Conference Paper | Publisher: IEEE