Data Science Capstone Project
Title : “Improving Restaurant Reputation Using Yelp User Reviews” - Part 1: “Data Description and Data Wrangling”
Springboard Bootcamp, San Francisco
Reza Taeb
Spring 2018

**How to improve restaurant reputation even just one star using Yelp user reviews !**
**“ The path from being normal to good, and good to perfect !"**

**Data Description:**

The dataset used in this project is part of the dataset which is provided by Yelp for its round 10th challenge[1]. The whole dataset includes several datasets (business, checkins, photos, review, tip, user). Based on my capstone proposal and the attributes[2] of the mentioned sub datasets, for this project I just need two specific sub datasets, “review” and “business”. Both of these sub datasets have .json format.
The datasets have been downloaded from Yelp server on January 9, 2018 and kept locally during this project. I opened the review and business datasets separately as CSV format.

[1] https://www.yelp.com/dataset/challenge
[2] https://www.yelp.com/dataset/documentation/json

Data Science Capstone Project
Title : “Improving Restaurant Reputation Using Yelp User Reviews” - Part 1: “Data Description and Data Wrangling”
Springboard Bootcamp, San Francisco
Reza Taeb
Spring 2018

**Data Attributes:**

## Business

**Dimension :**

174567 * 15

**Memory Usage:**

20 MB

**Attributes:**

| Name | Description | Type |
|---|---|---|
| business_id | 22 character unique string business id | object |
| name | Business’s name | object |
| neighborhood | Business’s neighborhood | object |
| address | The full address of business | object |
| city | The city | object |
| state | 2 character state code | object |
| postal_code | The postal code | object |
| latitude | Latitude | float64 |
| longitude | Longitude | float64 |
| stars | The star rating (rounded to half-stars) | float64 |
| review_count | Number of reviews | int64 |
| is_open | 0 or 1 for closed or open, respectively | int64 |
| attributes | Business attributes to values | object |
| categories | An array of strings of business categories | object |
| hours | An object of key day to value hours | object |

## Review

**Dimension:**

5261669 * 9

**Memory Usage:**

361 MB

**Attributes:**

| Name | Description | Type |
|---|---|---|
| review_id | 22 character unique review id | object |
| user_id | 22 character unique user id | object |
| business_id | 22 character unique business id | object |
| stars | Star rating | int64 |
| date | Date format YYYY-MM-DD | object |
| text | The review itself | object |
| useful | Number of useful votes received | int64 |
| funny | Number of funny votes received | int64 |
| cool | Number of cool votes received | int64 |

Data Science Capstone Project
Title : “Improving Restaurant Reputation Using Yelp User Reviews” - Part 1: “Data Description and Data Wrangling”
Springboard Bootcamp, San Francisco
Reza Taeb
Spring 2018

By looking at the attributes of these two datasets and my goal in this project, initially I focused on some of these attributes:

| Attributes | Example | Attributes | Example |
|---|---|---|---|
| business_id | tnhfDv5Il8EaGSXZGiuQGg | name | Garaje |
| user_id | Ha3iJu77CxlrFm-vQRs_8g | state | CA |
| postal_code | 94107 | review_count | 1198 |
| stars | 4.5 | date | 2016-03-09 |
| categories | ["Mexican","Burgers","Gastropubs”] | attributes | {"RestaurantsTakeOut":true, "BusinessParking": {"garage": false,"street": true,"validated": false,"lot": false,"valet": false},} |
| text | Great place to hang out after work: the prices are decent, and the ambience is fun. It's a bit loud, but very lively. The staff is friendly, and the food is good. They have a good selection of drinks. | | |

- **Filtering “Restaurants” from our original dataset**

Since my project is just about the “Restaurants”, I have to separate the "Restaurant" category. I looked into the "categories" column to figure out what are the **words** that point out to restaurants and foods. For now, there is no "Null" entry for categories column, but I have to look more in depth to see if we have some untidy entries. Initially I divided the original business dataset into “restaurant” and “non_restaurant” subsets.
Glancing to the "categories" column for "non_restaurant" subset, I noticed another similar word to "Restaurant" which is **Food** . There are around 14500 observations that ***do not have RESTAURANT but have FOOD as one of their categories tags.*** So I need to look at this tag (Food) more carefully. However I need to focus more on the entries that have "Food" as one of their categories, since some of them may provide some products or services that are related to the food (such as groceries, etc.) and **Not Serving Food** .
Looking at the part of the data that have **Food** and not **Restaurant** I figured out that majority of them have "Coffee and Tea" , "Grocery", "Ice Cream and Frozen Yogurt", "Bakeries", "Convenience Store" , and so on. So let's just keep the ones that have **Restaurant**, since there are enough observations (more than 54 K) and keeping the ones with **Food** may make it very

complex and may bring some biases to our final analysis on reviews (for example the businesses that serve Italian ice cream or american coffee may affect our analysis of whole Italian or American restaurants group).

- **Filtering “Non_English” reviews**

My project is based on analysing user reviews (texts). Yelp is a globally used platform and the downloaded dataset includes non_english texts that are needed to be filtered. I did the filtering based on the “state” column of dataset.

- **Adding “Country” column**

Using “state” column of original dataset, I added another column “Country” for future analysis. Separating original dataset into 3 English spoken countries (USA, Canada, Britain - including UK, Scotland - ) may expand our insight of this dataset.

- **Filtering 7 most popular types of Food (Adding “food_type” column)**

In this project, I also wondered whether type of food (Italian, American, Mexican, etc.) may affect factors or their weights that influence customers’ text reviews and stars (ratings). Therefore, I checked “Categories” column and add another column (“food_type”). Since there are some restaurants that serve different types of food at the same time, I just filtered the ones that are specifically serve one of these 7 most popular types of food to avoid any complexity and bias that may combination of foods may bring into our analysis.

Eventually I came up with the final version of dataset , “**df_rest_eng_top7**” , which consists of ***English spoken restaurants which serve one of the top 7 popular types of food*** (American, Italian, Mexican, Chinese, Japanese, Indian, and Thai) based on the original dataset.

| | |
|---|---|
| American | 8759 |
| Italian | 3927 |
| Mexican | 3652 |
| Chinese | 3382 |
| Japanese | 1993 |
| Indian | 1231 |
| Thai | 999 |

Number of restaurants based on each type of food

Our final dataset has 23943 entries.
Since the ultimate goal of this project is coming up with the model that can predict the rating of a restaurant based on its user reviews and type of food, I had to check whether there are enough observations (number of restaurants and review counts) for any type of food and

stars, as you can see in the below tables there are almost enough observations for each food type and star.

| food_type | stars | count | mean | sum |
|---|---|---|---|---|
| American | 1.0 | 21 | 4.952381 | 104.0 |
| | 1.5 | 104 | 14.798077 | 1539.0 |
| | 2.0 | 408 | 25.960784 | 10592.0 |
| | 2.5 | 993 | 48.657603 | 48317.0 |
| | 3.0 | 1877 | 61.259457 | 114984.0 |
| | 3.5 | 2352 | 92.075680 | 216562.0 |
| | 4.0 | 2064 | 160.362888 | 330989.0 |
| | 4.5 | 792 | 137.657828 | 109025.0 |
| | 5.0 | 148 | 22.858108 | 3383.0 |
| Chinese | 1.0 | 19 | 6.421053 | 122.0 |
| | 1.5 | 41 | 8.634146 | 354.0 |
| | 2.0 | 191 | 20.471204 | 3910.0 |
| | 2.5 | 414 | 29.120773 | 12056.0 |
| | 3.0 | 865 | 36.114451 | 31239.0 |
| | 3.5 | 985 | 59.367513 | 58477.0 |
| | 4.0 | 661 | 63.682300 | 42094.0 |
| | 4.5 | 176 | 45.812500 | 8063.0 |
| | 5.0 | 30 | 6.533333 | 196.0 |
| Indian | 1.0 | 2 | 3.000000 | 6.0 |
| | 1.5 | 14 | 10.642857 | 149.0 |
| | 2.0 | 27 | 11.407407 | 308.0 |
| | 2.5 | 101 | 13.188119 | 1332.0 |
| | 3.0 | 207 | 22.942029 | 4749.0 |
| | 3.5 | 343 | 49.629738 | 17023.0 |
| | 4.0 | 340 | 67.823529 | 23060.0 |
| | 4.5 | 173 | 51.919075 | 8982.0 |
| | 5.0 | 24 | 6.500000 | 156.0 |
| Italian | 1.0 | 28 | 7.321429 | 205.0 |
| | 1.5 | 89 | 8.910112 | 793.0 |
| | 2.0 | 182 | 15.406593 | 2804.0 |
| ... | | ... | ... | ... |
| | 4.0 | 1090 | 93.892661 | 102343.0 |
| | 4.5 | 456 | 77.296053 | 35247.0 |
| | 5.0 | 82 | 12.195122 | 1000.0 |
| Japanese | 1.0 | 6 | 6.000000 | 36.0 |
| | 1.5 | 10 | 20.900000 | 209.0 |
| | 2.0 | 48 | 28.416667 | 1364.0 |
| | 2.5 | 135 | 37.155556 | 5016.0 |
| | 3.0 | 358 | 54.519553 | 19518.0 |
| | 3.5 | 572 | 94.321678 | 53952.0 |
| | 4.0 | 583 | 135.560892 | 79032.0 |
| | 4.5 | 244 | 140.856557 | 34369.0 |
| | 5.0 | 37 | 16.378378 | 606.0 |
| Mexican | 1.0 | 15 | 6.266667 | 94.0 |
| | 1.5 | 65 | 15.630769 | 1016.0 |
| | 2.0 | 193 | 24.264249 | 4683.0 |
| | 2.5 | 476 | 37.873950 | 18028.0 |
| | 3.0 | 749 | 49.193591 | 36846.0 |
| | 3.5 | 888 | 80.278153 | 71287.0 |
| | 4.0 | 806 | 107.406948 | 86570.0 |
| | 4.5 | 375 | 82.040000 | 30765.0 |
| | 5.0 | 85 | 16.494118 | 1402.0 |
| Thai | 1.0 | 2 | 4.500000 | 9.0 |
| | 1.5 | 13 | 6.307692 | 82.0 |
| | 2.0 | 46 | 11.217391 | 516.0 |
| | 2.5 | 60 | 24.550000 | 1473.0 |
| | 3.0 | 134 | 27.552239 | 3692.0 |
| | 3.5 | 274 | 77.554745 | 21250.0 |
| | 4.0 | 335 | 125.495522 | 42041.0 |
| | 4.5 | 117 | 112.051282 | 13110.0 |
| | 5.0 | 18 | 8.055556 | 145.0 |

Number of restaurant (count) and number of reviews (sum) for each type of food and star

This is the general info of our final dataset:

```
Data columns (total 17 columns):
business_id     23943 non-null object
name            23943 non-null object
neighborhood    10073 non-null object
address         23857 non-null object
city            23943 non-null object
state           23943 non-null object
postal_code     23917 non-null object
latitude        23943 non-null float64
longitude       23943 non-null float64
stars           23943 non-null float64
review_count    23943 non-null float64
is_open         23943 non-null float64
attributes      23943 non-null object
categories      23943 non-null object
hours           23943 non-null object
country         23943 non-null object
food_type       23943 non-null object
```

I also took a deep dive into the dataset and found out that there is no missing value for the columns that I need in my analysis (I do not need “neighborhood” column in my analysis).
Review dataset also consists of texts, so there is no missing value in the datasets.

The main columns which I used in this project are review texts, star ratings and food type, so there is no outliers or at least there is no obvious outlier at this stage of project.