Project Report

Yelp Recommendation system for new restaurants

CMPE 256 - Large Scale Analytics
Summer 2019

Submitted by
Akshata Kulkarni – 013829010

Under the guidance of
Prof. Shih Yu Chang

Akshata.Kulkarni@sjsu.edu

https://github.com/akshatakulkarni98/Yelp_Recommendation_system_NewRestaurants

1. Abstract

Yelp has a vast database of reviews, ratings, and general information provided by the community about businesses. Yelp provides consumers with a myriad of options and information even when searching for an especially specific service or goods. The new businesses added to the database, which have lesser number of ratings and reviews might not get recognized in this huge amount of data. In this project, a recommendation system will be developed for new restaurants added to the Yelp database. A recommendation system will help Yelp to learn the user's likes and dislikes. Based on the information about the restaurants and ratings given by Yelp users to various restaurants, we can predict the rating of a particular user for a new restaurant and recommend using content-based filtering techniques. In this project we have explored two different ways to recommend the new restaurants to the user who might like them.

2. Introduction

Yelp is a business directory service and crowd-sourced review forum. It contains review data of various businesses in the city and it can be helpful for users in choosing a suitable service for them. Yelp has gained lot of popularity in short amount of time and received massive amount of user and business data. Because of overwhelming number of reviews, it is impossible for a single user to get information about suitable services according to their needs. Recommender system is the solution for this problem. Recommender system will not only help users to find the products/services, it will also be beneficial for businesses, to attract potential customers.

2.1 Problem Statement

A recommendation system predicts the user review rating for a list of unvisited restaurants based on user's personal preferences and personalities. This project is to recommend new restaurants to potential customers residing in San Francisco Bay area.

Recommendation system is mainly implemented using two techniques. Collaborative filtering technique and Content based filtering. Collaborative filtering approach, builds a model from a user's past history as well as similar decisions made by other users; then use that model to predict items that the user may be interested in. Content-based filtering is using the technique to analyze a set of documents and descriptions of items previously rated by a user, and then build a profile or model of the user's interests based on the features of those rated items. Using the profile, the recommender system can filter out the suggestions that would fit for the user.

Collaborative filtering technique needs sufficient review data in the form of ratings for items that user is interested in. So, this technique suffers from a cold start problem. Cold start problem can be relevant both for new users and for a new product, which has yet no reviews or history of being a success among a certain group of users. As scope of our project is to give recommendations for new restaurants, this technique will not help to solve our problem.

On the other hand, content-based filtering technique is the solution for item cold start problem. For recommendation in content-based filtering, content of the item is used to predict users rating. So, the new items can be recommended effectively. In this project we will be solving this problem with help pf two different ways of content-based filtering techniques.

3. Data Extraction

The scope of the project is to provide new restaurants recommendation to the users residing in the San Francisco Bay Area. Yelp provides an academic dataset that has clean data about the business, reviews and users. But this dataset does not contain any data of San Francisco Bay Area. So, in order to solve this problem, the data had to be scrapped from Yelp using Fusion APIs.

The Yelp Fusion APIs are RESTful APIs and users can retrieve business review and rating, information for a particular geographic region or location, get review information for a particular business and track recent reviews for a particular business.

The Yelp Fusion API uses private key for authentications and authorizations. The private API Key for users will be automatically generated after creating app. The default output is JSON. The following Yelp Fusion APIs are available: Search, Phone Number Search, Business Search, Transaction, Reviews, and AutoComplete - each API has a separate Programmable Web entry.

Endpoints used in this project:

1. /businesses/search
2. /businesses/
3. /businesses/{id}/reviews

For this project, parameters are set as mentioned below.

Parameter name	Value
term	restaurants
location	San Francisco
radius	40000
limit	50

offset	50,100...1050

Table 3.1

The data from Fusion is enforced with rate limiting by Yelp and this resulted in scrapping the data multiple times. The data gathered by this activity resulted in approximately 2000 restaurants and 6000 user's reviews.

Please refer to the below code snippet as an example.

```
import requests

## Example for /businesses/{id}
## Get business details of a given business_id
## https://www.yelp.com/developers/documentation/v3/business

BASE_BUSINESS_URL = "https://api.yelp.com/v3/businesses/"
business_ids = ['uYHaNptLzDLoV_JZ_MuzUA']

for business_id in business_ids:
    response = requests.get(
        'https://api.yelp.com/v3/businesses/{}'.format(business_id),
    headers={'Authorization': 'Bearer {}'.format(api_key)})
    print ("REQUEST " + str(response.request.url) + "\n")
    json_response = response.json()
    print (json_response)
```

```
REQUEST https://api.yelp.com/v3/businesses/uYHaNptLzDLoV_JZ_MuzUA

{'id': 'uYHaNptLzDLoV_JZ_MuzUA', 'alias': 'motel-one-edinburgh', 'name': 'Motel One', 'image_url': 'https://s3-media
2.fl.yelpcdn.com/bphoto/EKpZ2LrTZhq6Zunn2XMrlw/o.jpg', 'is_claimed': False, 'is_closed': False, 'url': 'https://www.y
elp.com/biz/motel-one-edinburgh?adjust_creative=4h-5kJ2Iz88OGBpTkesDfg&utm_campaign=yelp_api_v3&utm_medium=api_v3_bus
iness_lookup&utm_source=4h-5kJ2Iz88OGBpTkesDfg', 'phone': '', 'display_phone': '', 'review_count': 20, 'categories':
 [{'alias': 'hotels', 'title': 'Hotels'}], 'rating': 4.0, 'location': {'address1': 'Market Street', 'address2': '',
 'address3': '', 'city': 'Edinburgh', 'zip_code': 'EH1', 'country': 'GB', 'state': 'EDH', 'display_address': ['Market
Street', 'Edinburgh EH1', 'United Kingdom'], 'cross_streets': ''}, 'coordinates': {'latitude': 55.9508705, 'longitud
e': -3.1913099}, 'photos': ['https://s3-media2.fl.yelpcdn.com/bphoto/EKpZ2LrTZhq6Zunn2XMrlw/o.jpg', 'https://s3-media
3.fl.yelpcdn.com/bphoto/B2-ROr0D6YMYnE9GdXVo3w/o.jpg', 'https://s3-media2.fl.yelpcdn.com/bphoto/D06CabcjTSjXFmnz4nN-8
Q/o.jpg'], 'price': '£', 'transactions': []}
```

Fig 3.1

4. Data Analysis

The restaurant and review data were extracted using two different Fusion APIs and later clubbed together as a single entity for better understanding and implementation. This scrapped data consists of restaurant name, address, categories, rating, review count, reviews, user_id and other raw information. Not all of this would be helpful for implementation and analysis. Hence the useful data (restaurant id, name, categories, rating, reviews, user id) is extracted and written on to JSON file. This will help in implementation, analysis and validation.

During Data Analysis, an attempt was made to get the bird's eye view of the data. So, the line graph for category and total number of restaurants was plotted. Similarly, the relation between review rating and the number of restaurants was identified in the San Francisco Bay Area. Please refer to the below graphs for data visualization for San Francisco Bay Area.

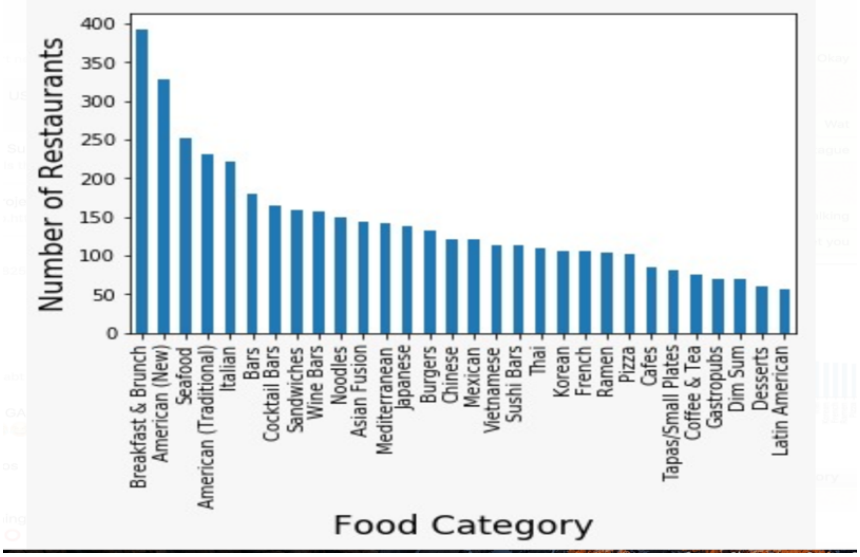

Fig 4.1

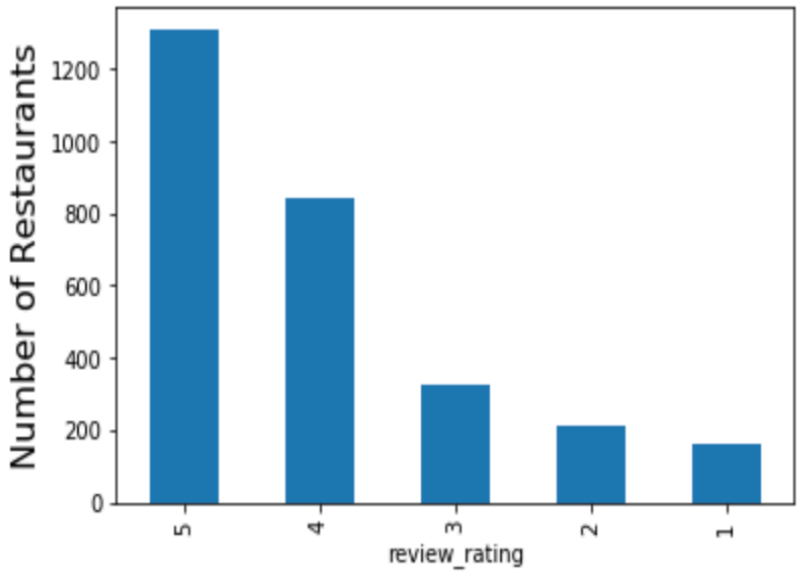

Fig 4.2

5. Data Preprocessing

Data preprocessing is the data mining technique to transform the raw real-world data into a meaningful format. Usually the data collected from real world will be noisy, incomplete and inconsistent. In the data pre-processing step, data will be cleaned and modified according to the project needs.

Data extracted from Yelp was in JSON format. The business data has many information related to the restaurants like id, name, categories, coordinates, phone_no,

distance, image_url, price, rating, review_count, review_list, transaction and url. Only useful data is extracted from JSON file onto CSV file as shown in the screenshot below.

name	business_id	categories	is_closed	review_id	review_rating	review_text	user_id
Fog Harbor Fish House	f-m7-hyFzkf0HSEeQ2s-9A	Seafood,Bars	False	eS_qxC-qC8RnJY-FXD3djg	5	The restaurant is in a perfect location. You m...	3UxhJ0zdww8ptAESBMwAaQ
Fog Harbor Fish House	f-m7-hyFzkf0HSEeQ2s-9A	Seafood,Bars	False	vmY1p68iTuh04iFm8ovglA	3	My girlfriend and I flew into SFO and decided ...	XoPZFbhk60YaXOlFvfBFIA
Fog Harbor Fish House	f-m7-hyFzkf0HSEeQ2s-9A	Seafood,Bars	False	c-TXm5KzuSO3TrzZ4AAsCA	1	Ambiance is nice but this is another tourist t...	iq9b1xt3PG1rBbjGY4l6AQ
Marufuku Ramen SF	HHtpR0RslupSQ99GIlwW5A	Ramen	False	mYK-uNIT2DgiTpq1QO-CZQ	5	This is one of my favorite ramen places in the...	ZDLnBs5w9Lh_j2t9fBLRLA

Fig 5.1

5.1 Feature Extraction

1. One hot encoding:

There are many keywords in the categories section for each restaurant. To implement content-based recommendation system we need distinct keywords to understand the features/cuisines of that restaurant. Please refer to the business_category_mapping.csv for one hot encoded matrix for each restaurant.

	Wine & Spirits	Acai Bowls	Afghan	American (New)	American (Traditional)	Arabian	Arcades	Argentine	Armenian	Asian Fusion	...	Vegan	Vegetarian	Venezue
f-m7-hyFzkf0HSEeQ2s-9A	0	0	0	0	0	0	0	0	0	0	...	0	0	
HHtpR0RslupSQ99GIlwW5A	0	0	0	0	0	0	0	0	0	0	...	0	0	
ZoZjbOYR-apY8XvommlNUA	0	0	0	0	0	0	0	0	0	1	...	0	0	
gqVl3RprESEqklPeJH0yOg	0	0	0	0	0	0	0	0	0	0	...	0	0	
8kck3-K4zYKTJbJko0JlXQ	0	0	0	0	0	0	0	0	0	0	...	0	0	

Fig 5.1.1

2. Restaurant rating matrix:

Each user in our data has a given rating to some restaurants. The matrix that has restaurant_id versus user_id and containing ratings R(i ,j) as value was required for the implementation. Please refer to the User_restaurant_rating.csv file for this matrix.

	f-m7-hyFzkf0HSEeQ2s-9A	HHtpR0RslupSQ99GIlwW5A	ZoZjbOYR-apY8XvommlNUA	gqVl3RprESEqklPeJH0yOg	8kck3-K4zYKTJbJko0JlXQ	eYXwVR4mMAjzk
3UxhJ0zdww8ptAESBMwAaQ	5.0	NaN	NaN	NaN	NaN	
XoPZFbhk60YaXOlFvfBFIA	3.0	NaN	NaN	NaN	NaN	
iq9b1xt3PG1rBbjGY4l6AQ	1.0	NaN	NaN	NaN	NaN	
ZDLnBs5w9Lh_j2t9fBLRLA	NaN	5.0	NaN	NaN	NaN	

Fig 5.1.2

6. Technical Implementations

For this project, I have utilized content-based filtering techniques to implement the recommendation system for new restaurants. Unlike Collaborative filtering technique, content-based technique doesn't involve other users rating for recommendation. Based on what we like, the algorithm will simply pick items with similar contents to recommend.

In this case there will be less diversity in the recommendations, but this will work without other user's information. So, this technique is best suited for our case as we don't have any user's data for new restaurants. Here I have implemented content-based filtering techniques in 2 different approaches.

6.1 Content based filtering - Approach 1:

The idea here is to find the similarity between all pairs of items (between restaurants), then recommend the most similar items to users who already rated those similar items. For example, If the new restaurant is a Mexican restaurant, then it will be recommended to users who have liked other similar Mexican restaurants. The similarity of restaurants will be derived from the description of the restaurants. In our data set, we have categories column which gives the description of restaurants.

6.1.1 Algorithm for Implementation:

Please refer to the algorithm used for the implementation.

Step 1: Combine all the keywords from title of restaurant and categories of restaurant.
Step 2: To calculate similarities, we need vectors. CountVectorizer function will be used to count each word in keyword column.
Step 3: Cosine Similarity will be used to calculate similarity between restaurants.
Step 4: Similarity scores will be sorted in ascending order. Top 10 most similar restaurants to the new restaurants will be obtained.
Step 5: Find the users who liked top 10 most similar restaurants and gave rating > 3. The new restaurants should be recommended to these users.

6.1.2 Results:

1. **Similar restaurants:** To test the implementation of recommendation system, I have created synthetic data of 10 new restaurants (a1 – a10) with different categories and no user review information. information. The results of the implementation were satisfactory.

```
find_similar_restaurant(Similar_restaurants('a6'))

Out[46]: {'Asian Box',
          'Barnzu',
          'Bon Voyage',
          'Champa Garden',
          'Osha Thai',
          "Rin's Thai Restaurant",
          'Sanraku',
          'Sushirrito',
          'Wicked Star',
          'pomelo on Judah'}
```

Fig 6.1.2

2. **Recommend new restaurant to users:**

```
In [37]: # List of users who might like res_id
         high_star_rating_users_who_may_like

Out[37]: {'1gdZQ47zus0DsW-F9hdGmQ',
          '3hSGiWeU55-t3ef_dsIqkw',
          'C3YcMYonAvBKZAD2uUnfig',
          'LflvReyrKC0fI63AWiP23g',
          'LljZPVdvobWxgbAaXDwM6Q',
          'MOgz0_VkT9AOJFQ5r33Y7A',
          'PrkYAAWHpcNLn0X1sVP2Ig',
          'RG91_Obi7yhHKAs5tUYgDQ',
          'ReoITf9K798Y_1aat7zh-Q',
          'SgUv6nrd1uKtDvppvOmP-A',
```

Fig 6.1.3

6.2 Content based filtering - Approach 2:

In this approach, user profile and item profile will be built using the ratings given by users in the past. these algorithms try to recommend items that are similar to those that a user liked in the past, or is examining in the present. To create a user profile, the system mostly focuses on A model of the user's preference and A history of the user's interaction with the recommender system.

Approach 2 leverages description or attributes from items the user has interacted to recommend similar items. It depends only on the user previous choices, making this method robust to avoid the cold-start problem. In this approach contents of the product are already rated based on the user's preference (User Profile), while the genre of an item is an implicit feature that it will be used to build Item Profile. An item score is then predicted by using both profiles and recommendation can be made. Similar to approach 1, TF-IDF technique will also be used in this approach.

6.2.1 Algorithm for Implementation:

Please refer to the algorithm used for the implementation.

Step 1: The data collected after feature extraction and data pre-processing are rating table and restaurant categories.

Step 2: User profile will be created with the available data to understand the user preferences (cuisine that user is interested).

Step 3: With the help of user preference and restaurant category, user to restaurant relationship will be derived.

Step 4: With help of TF*IDF, the relation between user and restaurant will be calculated and stored as matrix.

Step 5: Using this matrix, list of restaurants that a user might like can be predicted.

Step 6: And also given a restaurant id, list of users who might like can be predicted.

Result:

1. Recommendation for a particular user:

```
]: #List of business id that will be recommended to user_no : B-tnUR66MKSoLtzMrHBywA
recommender_restaurants_to_user('B-tnUR66MKSoLtzMrHBywA')
```

]:

business_id	B-tnUR66MKSoLtzMrHBywA	name	categories
QDb7LJMMJ9J4zwRC1LFRdg	37.916569	Bussaba	Thai,Gluten-Free,Comfort Food
AE2ong5QiDhG5bz4vQnZxQ	33.103460	Me & Tasty	Thai,American (New),Breakfast & Brunch
a5	30.202881	res_5	Cocktail Bars,American (New),Comfort Food
FWnV4Xuv1UP5-FGrr7aW4Q	30.202881	True Laurel	Cocktail Bars,American (New),Comfort Food
eOCVoXsOEnbr9c89RD0UEQ	29.498135	Pork Store Cafe	Breakfast & Brunch,Coffee & Tea,Comfort Food
9SsgoIxiQ7PHOfuG9P8QjQ	29.498135	Lunchpad	Sandwiches,Comfort Food,Breakfast & Brunch
0mNzmmh1mrdh5Cpg2QUBiw	24.405287	Lapisara Eatery	Breakfast & Brunch,Burgers,Thai
UGKNNQ6bjL-ZttHSdZISfA	24.405287	Blackwood	Thai,Breakfast & Brunch,Asian Fusion
ohWlEV89ljnbDXAHJM6_GQ	21.940345	ONE65 Bistro & Grill	American (New),French,Bars
WavvLdfdP6g8aZTtbBQHTw	21.940345	Gary Danko	American (New),French,Wine Bars

Fig 6.2.1

2. **Recommendations for new restaurant:** To test the implementation of recommendation system, I have created synthetic data of 10 new restaurants (a1 – a10) with different categories and no user review information. information. The results of the implementation were satisfactory.

```
# List of users for whom bus_id : a1 will be recommended
user_list = recommender_user_to_new_restaurants('a1')
user_list = user_list[['user_id']]
print(user_list)
```

```
                  user_id
690  J7L5EbWEHB8njkFjZkas7w
865  NtgSafaocv1HQ7DSzfRQLw
508  Dg8phi2yU6K_CczVbZNs4w
17   0F3ziFHw3WjfZR7ngN8M7A
858  NmcfLhYfNPmck_iHYCpBeA
505  DZaKmGqgDITCAX4HVFXUuw
110  2vYEopEQ2AfwyYnbmB9faQ
273  7ArWnVr7qfZd9rStpQKyXw
738  KCUFGSlK6lX7ZSlwTbtVZQ
684  IxSNLdIqzGu8AE1mwHhTKw
```

Fig 6.2.2

7. Conclusion

In this project I have addressed the issue of cold start problem which is very common in e-commerce website, new users and new items cold start problem. To solve new item cold start problem, I have made use of two different approaches to recommend new restaurants using content-based recommendation system. The testing was done using the synthetic dataset. I have added 10 new restaurants to the dataset manually which did not have any user ratings or reviews information. And recommended system output was as expected.

As future scope to improve the accuracy of the recommendation system, we can make use of review text given by users and apply NLP techniques to extract the features.

And also, Content-based method provides a limited degree of novelty, since it has to match up the features of profile and items. It suffers from over-specialization problem. To avoid this problem, we can build a hybrid recommender system with content-based and collaborative filtering techniques.

8. References

1. https://www.yelp.com/developers/documentation/v3/get_started
2. https://pdfs.semanticscholar.org/7348/48c5f6e6a6cb079da14d4f68159681ab44d5.pdf
3. https://arxiv.org/pdf/1805.09023.pdf