Capstone Project – Report
Andrea Tamburri

1. Business Problem

1.1 Introduction

The total value of the global travel and tourism industry is one of the world's largest industries with a global economic contribution of over 7.6 trillion U.S. dollars.
This is a very complex and heterogeneous market, with tons of different businesses all around the world, as, for example:

- Aribnb: an online marketplace for arranging or offering lodging, homestays or tourism experience, with revenues of over 2.6 billion
- Booking.com: a metasearch engine for lodging reservation, with total revenues of over 14.5 billion
- Ryanair: a competitive cost airline, with total revenue amount of over 7.2 billion
- Yelp: a business directory service and crowd-sourced review forum, with total revenue amount of 1 billion

Those are just few examples of what the tourism market as a whole can offer, and how can be remunerative.

1.2 Business Problem

The main idea of the whole project that it's going to be developed, is to leverage the information obtained by Foursquare about venues to cluster some of the major cities all around the world in order to create a **content based recommender system:** for example, talking about Airbnb, **if a customer decide to rent an apartment in Paris, and he liked it, he is possibly willing to rent apartments in cities similar to Paris** (in our case similarity is "just" given by venues distribution) that will be offered in the main page. **This system improves the customer experience, increasing their purchases or, in the other way around, increasing the revenue of the company.**
In this project I'll focus on "selling" them the "cluster matrix" of the whole recommender system.
Broadly speaking, in a real case scenario, I'd probably go for one of the two option:

- Selling to the companies a more complete cluster matrix, in order to have a broader point of view on each location and exploit in dept the advantages the reccommender systems: different clusters, based on, for example, venues distribution (the one that'll do), geolocation, life cost, etc.
- Selling, as a Saas, the whole recommender system: in this case, the cluster matrix as to be integrated with the information about customers behaviours (e.g., taking always Airbnb as example, the ratings about their experience in different localities) in order to create the "user profile" and, finally, in combination with the cluster matrix, the "recommendation matrix" that will be used to recommend to that customer the locations with the highest scores.

2. Data acquisition and cleaning

2.1 Data source

There are three major data sources used for this project:

- A Yelp businesses revue dataset, containing veues reviews from all around the world
- The "World Cities Table", in which you can find the geolocation of the most important cities from all around the world
- Foursquare, in which we can find more information about venues location, category, ratings

Yelp Dataset

A great repository of venues reviews, geolocation and users that actually made the reviews can be easily found in two Kaggle dataset that you can find here.
In our case this consists on two datasets:

- *Yelp_academic_dataset_business.json*
- *Yelp_academic_dataset_review.json*

The first one contains information about venues reviewed, as the address, the city in which is it located, the average rating etc.

```
RangeIndex: 192609 entries, 0 to 192608
Data columns (total 14 columns):
address         192609 non-null object
attributes      163773 non-null object
business_id     192609 non-null object
categories      192127 non-null object
city            192609 non-null object
hours           147779 non-null object
is_open         192609 non-null int64
latitude        192609 non-null float64
longitude       192609 non-null float64
name            192609 non-null object
postal_code     192609 non-null object
review_count    192609 non-null int64
stars           192609 non-null float64
state           192609 non-null object
```

Table1: Yelp_academic_dataset_business.json structure

The dataset has been cleaned of all the useless columns in order to have just the *business_id* and the *city* of the business. The purpose of the cleaning is because we only need to connect the information about the city of the venue with the review made for that venue.

The latter can be found in the second dataset, the *yelp_academic_dataset_review.json.*
This table contains the stored reviews of the previous businesses: the original dataset was bigger than 5GB, so it was impossible to upload entirely on IBM Watson Studio. In order to upload as much as possible, the dataset has been divided on chunks of 500K rows each. The total dataset uploaded contains 5M records of businesses reviews.

business_id	cool	date	funny	review_id	stars	text	useful	user_id
ujmEBvifdJM6h6RLv4wQIg	0	2013-05-07 04:34:36	1	Q1sbwvVQXV2734tPgoKj4Q	1	Total bill for this horrible service? Over $8G...	6	hG7b0MtEbXx5QzbzE6C_VA
NZnhc2sEQy3RmzKTZnqtwQ	0	2017-01-14 21:30:33	0	GJXCdrto3ASJOqKeVWPi6Q	5	I *adore* Travis at the Hard Rock's new Kelly ...	0	yXQM5uF2jS6es16SJzNHfg
WTqjgwHlXbSFevF32_DJVw	0	2016-11-09 20:09:03	0	2TzJjDVDEuAW6MR5Vuc1ug	5	I have to say that this office really has it t...	3	n6-Gk65cPZL6Uz8qRm3NYw
ikCg8xy5JIg_NGPx-MSIDA	0	2018-01-09 20:56:38	0	yi0R0Ugj_xUx_Nek0-_Qig	5	Went in for a lunch. Steak sandwich was delici...	0	dacAIZ6fTM6mqwW5uxkskg
b1b1eb3uo-w561D0ZfCEiQ	0	2018-01-30 23:07:38	0	11a8sVPMUFtaC7_ABRkmtw	1	Today was my second out of three sessions I ha...	7	ssoyf2_x0EQMed6fgHeMyQ

Table2: *Yelp_academic_dataset_reviews.json* raw table

The final table has been made joining the two Yelp datasets in order to put together all the necessary information to build our content-based recommender system (in particular this table we'll let us build the user's profile table: the one that explains how much the given user likes that cluster of cities).

user_id	city	stars_x
hG7b0MtEbXx5QzbzE6C_VA	Las Vegas	1
RBXSJA372ilErzNwz0jXvQ	Las Vegas	4
x3brMMbJrAW9PwW5A6YL5w	Las Vegas	1
Skzdl0sWhW88525a1vr59g	Las Vegas	1
3Y25VDfnQVcuc33T-U3Z6A	Las Vegas	5

Table3: final table extracted from Yelp dataset

The table contains just the information about the review's ratings made by the users in the city of the business. **That's because we are assuming that the city's rating of one particular user, is the mean of all the ratings made in that city.** The last step of preprocessing is to obtain for each user the mean rating of each city he reviewed.

user_id	city	stars_x
---1lKK3aKOuomHnwAkAow	Henderson	3.666667
---1lKK3aKOuomHnwAkAow	Las Vegas	3.934211
---1lKK3aKOuomHnwAkAow	North Las Vegas	3.666667
---94vtJ_5o_nikEs6hUjg	Phoenix	5.000000
---PLwSf5gKdIoVnyRHgBA	Phoenix	5.000000

Table4: Final table with mean calculated grouping by *user_id* and *city*

World Cities Table

Information about city geolocation can be easily found from the web: check the website at https://simplemaps.com/data/world-cities this interesting table about the biggest cities from all around the world.

As you can see below, this table, include:

- City name
- Latitude
- Longitude
- Country of the city
- Population of the city
- Etc.

city	city_ascii	lat	lng	country	iso2	iso3	admin_name	capital	population	id
Tokyo	Tokyo	35.6850	139.7514	Japan	JP	JPN	Tōkyō	primary	35676000	1392685764
New York	New York	40.6943	-73.9249	United States	US	USA	New York		19354922.0	1840034016
Mexico City	Mexico City	19.4424	-99.1310	Mexico	MX	MEX	Ciudad de México	primary	19028000	1484247881
Mumbai	Mumbai	19.0170	72.8570	India	IN	IND	Mahārāshtra	admin	18978000	1356226629
São Paulo	Sao Paulo	-23.5587	-46.6250	Brazil	BR	BRA	São Paulo	admin	18845000	1076532519
Delhi	Delhi	28.6700	77.2300	India	IN	IND	Delhi	admin	15926000	1356872604
Shanghai	Shanghai	31.2165	121.4365	China	CN	CHN	Shanghai	admin	14987000	1156073548
Kolkata	Kolkata	22.4950	88.3247	India	IN	IND	West Bengal	admin	14787000	1356060520
Los Angeles	Los Angeles	34.1139	-118.4068	United States	US	USA	California		12815475.0	1840020491

Table5: World Cities Table

The purpose of this table is to obtain the geolocation of the cities, in order to call the Foursquare API.
For computational matters in cloud (that's a capstone project made in the lite Watson Studio, with limited computational capacity), we'll take the top 30 most populated cities in the U.S. that match with the yelp dataset in order to cluster them through their venue distribution.
This will give us the cluster matrix (a table containing information about the clusters in which a given city is) that will later be combined with the user's profile in order to build the recommendation matrix.

Foursquare

We'll do some regular call at the Foursquare API to "explore" the venues around all the selected cities from the previous table.
The purpose is to obtain a table with a record for each venues of the selected cities, containing information about:

- Venue name
- Venue latitude and longitude
- **Venue category** (e.g. coffee shop or bank, **that uniquely describe a venue**)

The next step is to do some data preprocessing, like one hot encoding (in fact, k-means clustering doesn't match with categorical variable) for the venue category variable, substituted with a plethora of dummy variables.
The final step consists on extracting the mean frequency distribution of each venue category type for each city: this can be obtained grouping by city the table extracted from Foursquare.

City	ATM	Accessories Store	Adult Boutique	American Restaurant	Amphitheater	Antique Shop	Arcade	Arepa Restaurant	Argentinian Restaurant	...	Video Game Store	Video Store	Vietnamese Restaurant	Warehouse Store	Weight Loss Center	Wine Bar	Wine Shop	Wings Joint	Women's Store
Akron	0.0	0.0	0.0	0.026667	0.0	0.0	0.0	0.0	0.0	...	0.00	0.000000	0.0	0.0	0.0	0.013333	0.00	0.00	0.0
Allentown	0.0	0.0	0.0	0.020000	0.0	0.0	0.0	0.0	0.0	...	0.00	0.020000	0.0	0.0	0.0	0.000000	0.00	0.00	0.0
Antioch	0.0	0.0	0.0	0.043478	0.0	0.0	0.0	0.0	0.0	...	0.00	0.043478	0.0	0.0	0.0	0.000000	0.00	0.00	0.0
Aurora	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	...	0.00	0.000000	0.0	0.0	0.0	0.000000	0.00	0.00	0.0
Brooklyn	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.0	...	0.01	0.000000	0.0	0.0	0.0	0.000000	0.02	0.01	0.0

Table6: Cities venue types distribution table

Thanks to the preprocessing we'll be able to cluster the cities, via k-means clustering, based on their venue frequency distributions and, finally, obtain the **cluster matrix** that we want to "sell" or to exploit in order to build our content-based recommender system to the tourism companies, as stated in the **Business Problem paragraph** of the report. Of course is important to notice that, in order to have more sophisticated system, the cities should be clustered in many ways (e.g. life-cost or distance) in order to have a cluster matrix in which the cities are in more than just one cluster. In

this case we are focusing on exploiting the Foursquare API and get information from the venue distribution of each city.

3. Methodology

In this paragraph are presented the main components of the project:

- A data exploration part, in which the data are analyzed in order to get any useful insight and fully understand how those data are "good" to implement a recommender system
- A modelling part in which are explained the models and techniques used in order to implement the recommender system

3.1 Data Exploration

3.1.1 The most common venues for each city

In this section it's showed a table containing the most frequent venue types for each city, in order to get any insight on how many clusters could be done, and their composition.
Looking at the table, we can notice:

- An **"ethnic" cluster**, composed by cities (like Antioch, San Diego, Oakland) with lots of ethnic restaurants such as Italians, Mexicans, Chinese
- A **"fast food" cluster**, composed by cities (like Columbus or Phoenix) mainly containing fast food restaurants/sandwich places
- A "**bar" cluster**, composed by cities (like New York or Seattle, Cleveland) with bars, coffee shops, cafès as most frequent venues
- An **"outside experience" cluster**, composed by cities (like Los Angeles or Las Vegas) with lots of parks, trails, scenic lookout
- An **"others" cluster**, composed by cities with different venues from the rest of the cities like Greensboro or Henderson or Pittsburgh that have lots of Mobile phone shops/Repair Shops/Rental Car locations

city	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Akron	Sandwich Place	Bar	Café	Italian Restaurant	Bank
Allentown	Sandwich Place	Coffee Shop	Mexican Restaurant	Brewery	Gastropub
Antioch	Chinese Restaurant	Pharmacy	Intersection	Sports Bar	Sandwich Place
Aurora	Farm	Cosmetics Shop	Pet Store	Fish Market	Farmers Market
Brooklyn	Caribbean Restaurant	Pizza Place	Café	Mobile Phone Shop	Bakery
Charlotte	Pizza Place	Coffee Shop	Gym / Fitness Center	Sandwich Place	Park
Cleveland	Bar	Pub	Coffee Shop	Intersection	Convenience Store
Columbus	Fast Food Restaurant	Intersection	Fried Chicken Joint	Food Truck	Flower Shop
Concord	Pizza Place	Sandwich Place	Convenience Store	American Restaurant	Hotel
Dallas	Mexican Restaurant	Convenience Store	Pizza Place	Dive Bar	Dance Studio
Denver	American Restaurant	Pool	Pizza Place	Coffee Shop	Mexican Restaurant
Greensboro	Mobile Phone Shop	Sandwich Place	Clothing Store	Spa	Burger Joint
Harrisburg	Italian Restaurant	Café	Park	Gay Bar	French Restaurant
Henderson	Other Repair Shop	Auto Workshop	Auto Garage	Trail	Video Store
Lancaster	American Restaurant	Bar	Café	Sandwich Place	Coffee Shop
Las Vegas	Park	Moving Target	Trail	Food	Home Service
Los Angeles	Park	Scenic Lookout	Trail	Home Service	Lake
Madison	Coffee Shop	Hotel	Sandwich Place	Bar	New American Restaurant
Mesa	Sandwich Place	Cosmetics Shop	Grocery Store	Dance Studio	Convenience Store
New York	Bar	Coffee Shop	Mexican Restaurant	Pizza Place	Bakery
Oakland	Liquor Store	Chinese Restaurant	Mexican Restaurant	Park	Pizza Place
Ogden	Café	Mexican Restaurant	Ice Cream Shop	Italian Restaurant	Bar
Omaha	Park	Intersection	Café	Breakfast Spot	Hotel
Peoria	Grocery Store	Pizza Place	Sandwich Place	Liquor Store	Sushi Restaurant
Phoenix	Fast Food Restaurant	Convenience Store	Mexican Restaurant	Nail Salon	Sandwich Place
Pittsburgh	Hotel	Sandwich Place	Rental Car Location	Gym	Pizza Place
San Diego	Sandwich Place	Shipping Store	Middle Eastern Restaurant	Fast Food Restaurant	Deli / Bodega
Seattle	Coffee Shop	Bar	Cocktail Bar	Café	Yoga Studio

Figure1: Most frequent venues for each city

3.1.2 Average review rating per each city

The graph below plots the average review rating for each city, just grouping the reviews of the businesses by their city location.
Given the structure of the recommendation system content-based, that recommend the city with the highest score, this could help us to "predict" which cities could be the most likely to be recommended.

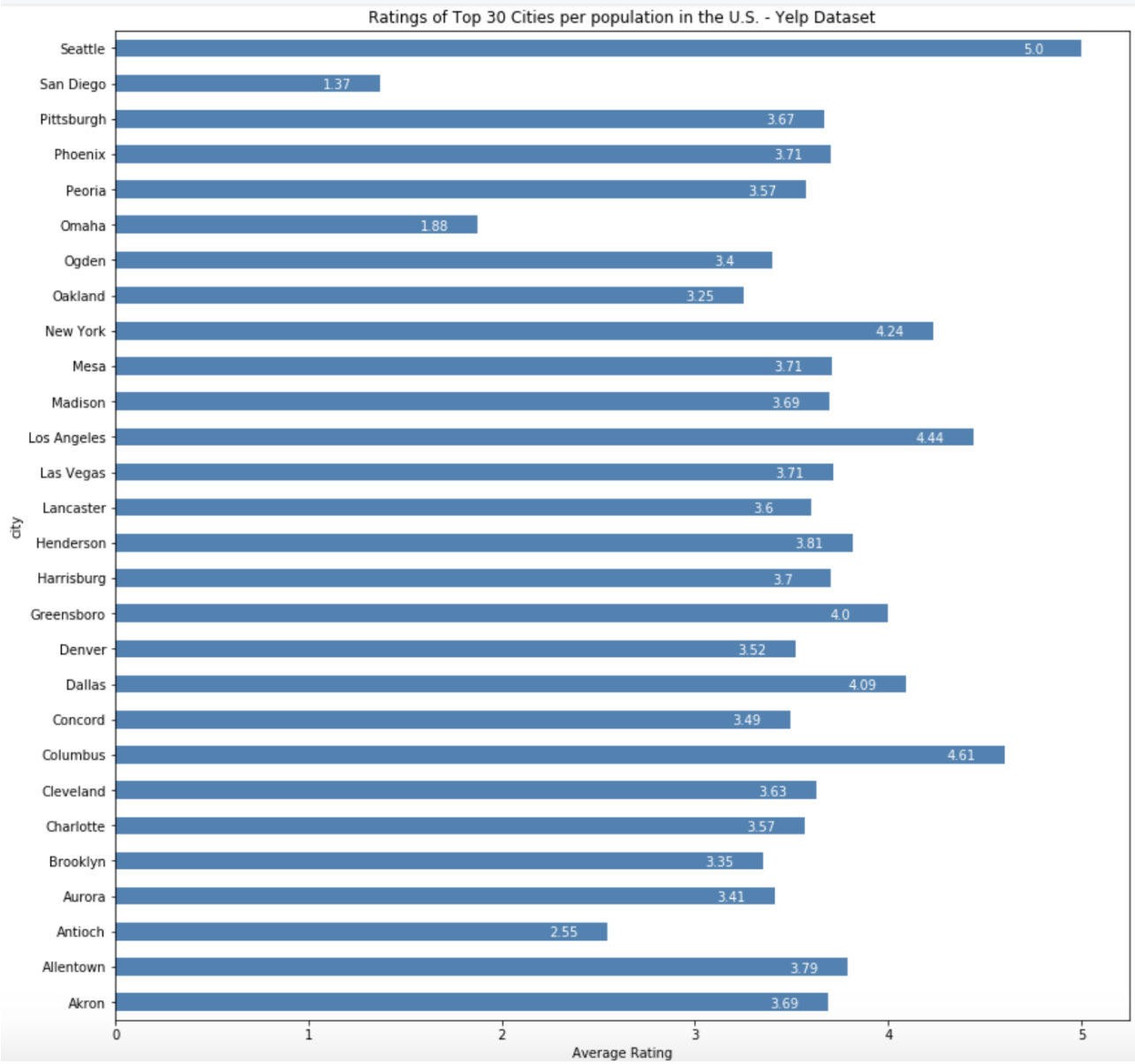

Figure2: Bar chart of the average rating per city

3.1.3 Pie chart of number of ratings per city

The graph below shows the ratings distribution among the cities to be clustered.
As we can see, the first three cities (Las Vegas, Phoenix and Charlotte) cover more than the 50% of all the reviews we have. This is not a good thing, since, in order to have a proper analysis of the average rating per city, we'd like to have an equal distribution for each city (e.g. Seattle's metric, with only three reviews, should be considered unreliable).

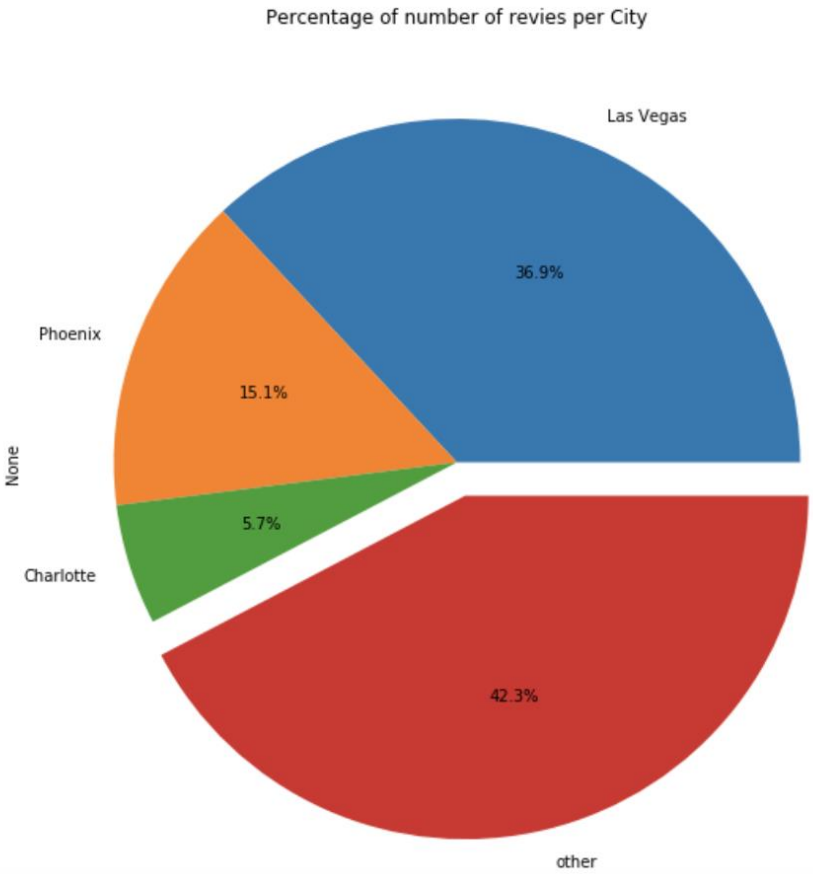

Figure3: Pie chart of the number of ratings for the top three cities against the others

3.1.4 Correlation between number of ratings and average rating per city

Since we can see that the three most rated cities have lower average rating compared to the cities with the lowest number or reviews (like Seattle, with just 3 reviews), could be interesting to see if there is any relationship between those two metrics in order to get if the dataset distribution of reviews could negatively impact the model we are going to do (since the rating could depend on how many reviews we have)
As we can see below, the scatterplot shows no correlation between those two metrics.

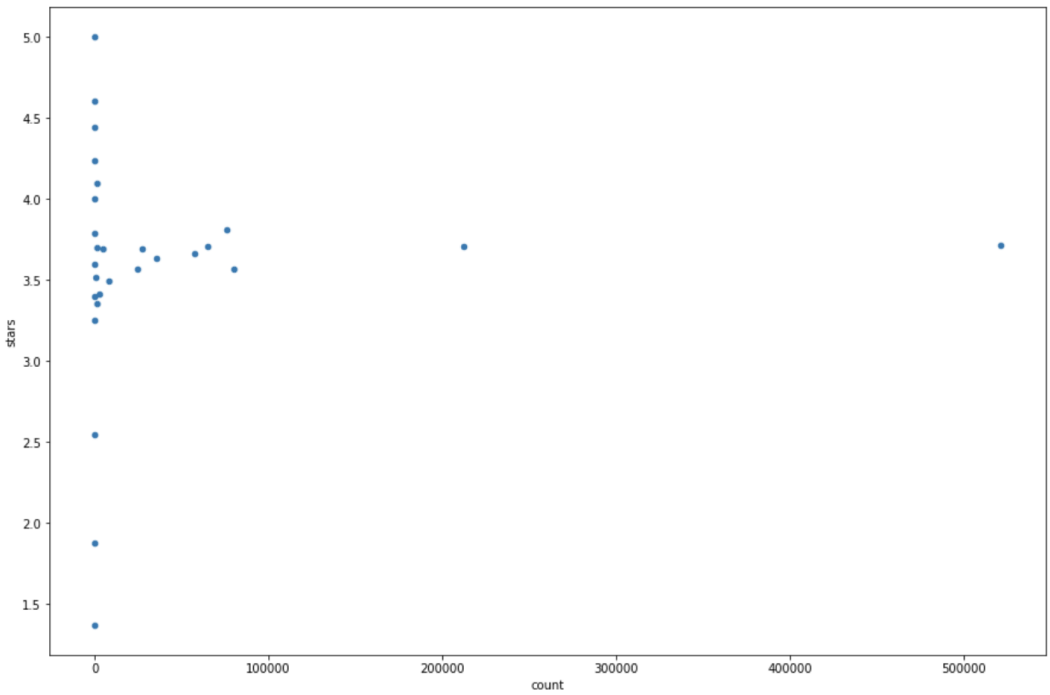

Figure4: Scatterplot of number of ratings vs average rating per city

3.2 Content-based recommendation system modelling

In order to build our content-based recommendation system, we need to do three main steps:

- **Creation of the city matrix**: a matrix n X m, with n the number of cities and m the number of clusters, and *aij=1* if the city *i* is included in the cluster *j* and vice versa. In our case, since we are clustering "just" for the venue distribution, each city will have one cluster.
- **Creation of User's profile**: a list, for each user, containing all the clusters he reviewed with the average rating per each cluster. This describe how much a user likes a cluster (really important for future recommendation)
- **Creation of recommendation matrix**: a matrix containing, for each user, the list of all the cities with their score, calculated in combination of the city matrix with the user's profile. The cities with the highest scores will be recommended to that user.

3.2.1 City Matrix via K-Means clustering

We are going to exploit the table containing information about venues frequency distribution for each city in order to cluster them, as already said.
The first thing to do is to detect the optimal number of cluster. In this case we'll use the elbow method:

- Elbow method via "distorsion" (It is calculated as the average of the squared distances from the cluster centers of the respective clusters. Typically, the Euclidean distance metric is used)
- Elbow method via "inertia" (It is the sum of squared distances of samples to their closest cluster center)

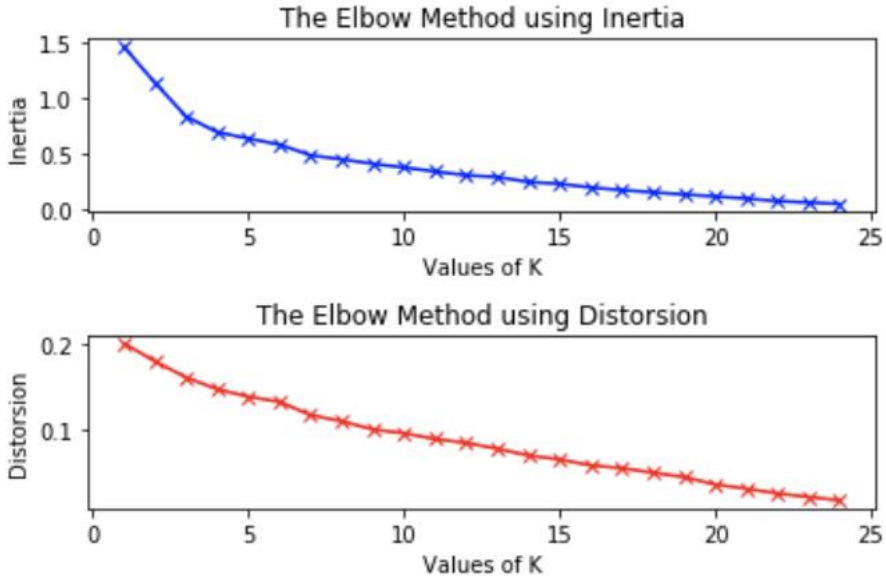

Figure5: Plots of different values of distorsion/inertia for different number of clusters K

To determine the optimal number of clusters, we have to select the value of k at the "elbow" ie the point after which the distortion/inertia start decreasing in a linear fashion.

Thus for the given data, Since the result is a bit ambiguous (there's an elbow using Inertia at K=3, but there's no strong evidence) I'll use a K=6, also because it's better to have more cluster for our recommender system, in order to be more accurate on the recommendation.

Now we can see the venues distribution of each cluster.

Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Bar	Coffee Shop	Café	American Restaurant	Sandwich Place
1	Park	Scenic Lookout	Trail	Home Service	Lake
2	Farm	Cosmetics Shop	Pet Store	Fish Market	Farmers Market
3	Pizza Place	Sandwich Place	Mexican Restaurant	Coffee Shop	Fast Food Restaurant
4	Park	Moving Target	Trail	Food	Home Service
5	Other Repair Shop	Auto Workshop	Auto Garage	Trail	Video Store

Figure6: venue distribution for each cluster

We can distinguish some of the clusters “already detected” like the “bar cluster” or the “fast food cluster” or the “others cluster” as well.

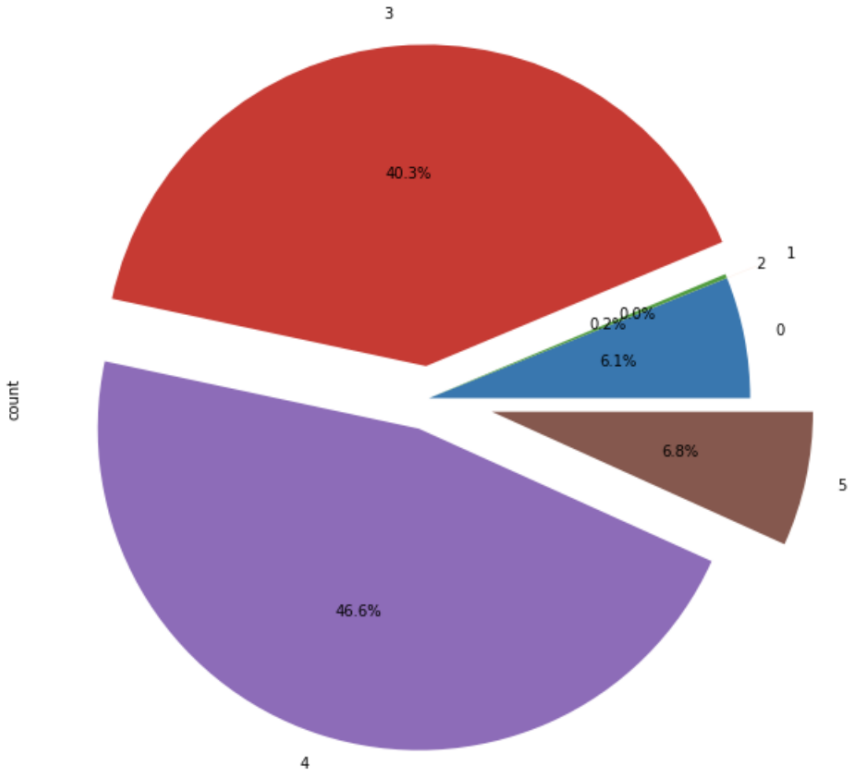

Figure7: pie chart displaying the number of reviews of each cluster

As we can see above, some clusters, as cluster 1 or 2, consists on just a bunch of reviews (43 and 2310 respectively), but we’ll keep the since there’s evidence on different venue distribution among them. Since the content-based recommendation system is mainly based on how the cities are clustered, I think is better to finely segment the cities, even if there will be few reviews, in order to give to the final consumer a better user-experience.

3.2.2 User's profile creation

The user's profile has been created as a Dataframe, containing the average rating for each cluster reviewed by an user.

For example, taking one random user, we can see:

user_id	Cluster Labels	stars_x
2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333
2LaXC_AW4I0EBU9FhzpOgg	2.0	3.500000
2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000
2LaXC_AW4I0EBU9FhzpOgg	4.0	3.968750
2LaXC_AW4I0EBU9FhzpOgg	5.0	4.500000

3.2.3 Recommendation matrix creation

This is the last step of our content-based recommendation system. The finale purpose of the entire process is to generate a list containing the scores of each city per every user, in order to recommend them the city with "their" highest score.

An example below:

	user_id	Cluster Labels	stars_x	city
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Denver
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Pittsburgh
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Columbus
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Peoria
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Antioch
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Greensboro
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Oakland
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Mesa
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Concord
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Allentown
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Charlotte
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Dallas
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Phoenix
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	San Diego
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Columbus
56293	2LaXC_AW4I0EBU9FhzpOgg	3.0	5.000000	Brooklyn
56295	2LaXC_AW4I0EBU9FhzpOgg	5.0	4.500000	Henderson
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Seattle
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Lancaster
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Lancaster
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Madison
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Harrisburg
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Akron
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Ogden
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Omaha
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	Cleveland
56291	2LaXC_AW4I0EBU9FhzpOgg	0.0	4.333333	New York
56294	2LaXC_AW4I0EBU9FhzpOgg	4.0	3.968750	Las Vegas
56292	2LaXC_AW4I0EBU9FhzpOgg	2.0	3.500000	Aurora

4. Results and discussion

Looking at the final result, described in the paragraph about creating the recommendation matrix, we can clearly see that there's no such a huge differentiation among the cities, as well as the number of cities is very low.
The differentiation between cities is mainly due to the clustering: in order to improve the model we should find more ways to cluster the cities, here few examples:

- Clustering by geolocation
- Clustering by cost
- Clustering by population
- Clustering by tourism traffic

In this way we are able to better identify each city, belonging to more than just one cluster.
This will improve the distinction between cities that, subsequently, will improve the distinction of the scores assigned to each city.
For the latter problem, it's clear that the model itself is perfectly suited for be scaled-up: this project has been run into a lite plan cluster on IBM Watson studio, with limited capabilities.
Scaling the computational capacity will lead us to include not only cities from U.S., but from all over the world.
The dataset itself is a shared "academic version" of the real Yelp dataset, so it's pretty straightforward that, in order to reach an optimal output, the entire basis should be scaled-up.

Another point of interest is the fact that a content-based recommendation system has some advantages and disadvantages:

- *Advantages:*
    - Learns user's preferences
    - Highly personalized for the user
- *Disadvantages:*
    - Doesn't take into account what others think of the item, so low quality item recommendations might happen
    - Extracting data is not always intuitive
    - Determining what characteristics of the item the user dislikes or likes is not always obvious

In order to minimize the disadvantaged, could be interesting implement an hybrid model containing some parts of the collaborative-filtering recommendation system, that, clustering the users instead of the cities, could minimize the problem of the content-based recommendation system.

5. Conclusions

In this project, I implemented a content-based recommendation system of U.S. cities, using the Yelp businesses reviews dataset.
The aim of this project is to build-up a system that, looking at the reviews made by an user, can recommend the best new city to explore.
This could be a competitive advantage for companies like Airbnb, Booking.com or Ryanair that can do some specific marketing campaigns and raise their probability to purchase and, subsequently, this will lead to an increase of revenues