Comparison of Neighbourhoods of Bengaluru, Seoul, Vancouver and San Francisco

Contents

Introduction

The problem I am considering is to compare the neighborhoods of four cities in four different countries. The countries I have selected are Bengaluru in India, Seoul in South Korea, Vancouver in Canada and San Francisco in USA. Attempt will be made to check which neighborhoods of the 4 cities are similar.

The target audience for this project is the owners of a restaurant chain which might already have their franchises set up in Vancouver and San Francisco and who want to enter new markets in Asia. They might consider other prominent tech cities in Asia since their target customer is the tech community. Since Seoul and Bengaluru have many MNCs and have a large tech community, they are the target of this project.

Data Sources

The neighborhood data of the four cities is taken from Wikipedia pages.

https://en.wikipedia.org/wiki/List_of_wards_in_Bangalore

https://en.wikipedia.org/wiki/List_of_districts_of_Seoul

https://en.wikipedia.org/wiki/List_of_neighbourhoods_in_Vancouver

https://en.wikipedia.org/wiki/List_of_neighborhoods_in_San_Francisco

Beautiful Soup library will be used to web-scrap from the Wikipedia pages. The geocode library to be used is geopy. Care is taken to limit the calls to be less than 1call/sec to meet the term of use of the library. Folium library will be used to represent the data on maps. And scikit-learn will be used to utilize machine learning. And Foursquare API will be used to gather neighborhood data.

Bengaluru has 199 neighbourhoods, Seoul has 25 neighourhoods, Vancouver has 33 and San Francisco has 114 neighbourhoods.

Importing Data

Importing data is divided into 3 stages. The first stage is getting list of neighbourhoods of the four cities from the above Wikipedia links. The second stage is getting location of neighbourhoods. The final stage is getting the venues in the neighbourhoods from Foursquare.

Neighbourhood lists of Cities

We already have the links of the Wikipedia pages from which we can get the list of neighbourhoods in each city. Beautiful Soup library is used to extract the information from the wikitables in the pages. This data is stored in a pandas dataframe. Along with the neighbourhoods, the city name, the state name and the country name are stored.

Geolocation of the Neighbourhoods

The geopy library is used to get the location data of the Neighbourhoods. Now in geopy library, Nominatim service is used. For using the free service of Nominatim, there is a restriction of 1call per sec to the service. To avoid ‘timeout’ error, there needs to be at least 1 sec gap in each call even if a for loop is used. To provide a sufficient gap to accommodate network delay, a gap of 2 sec is provided. The gap is provided by calling the sleep function.

Timeout Errors

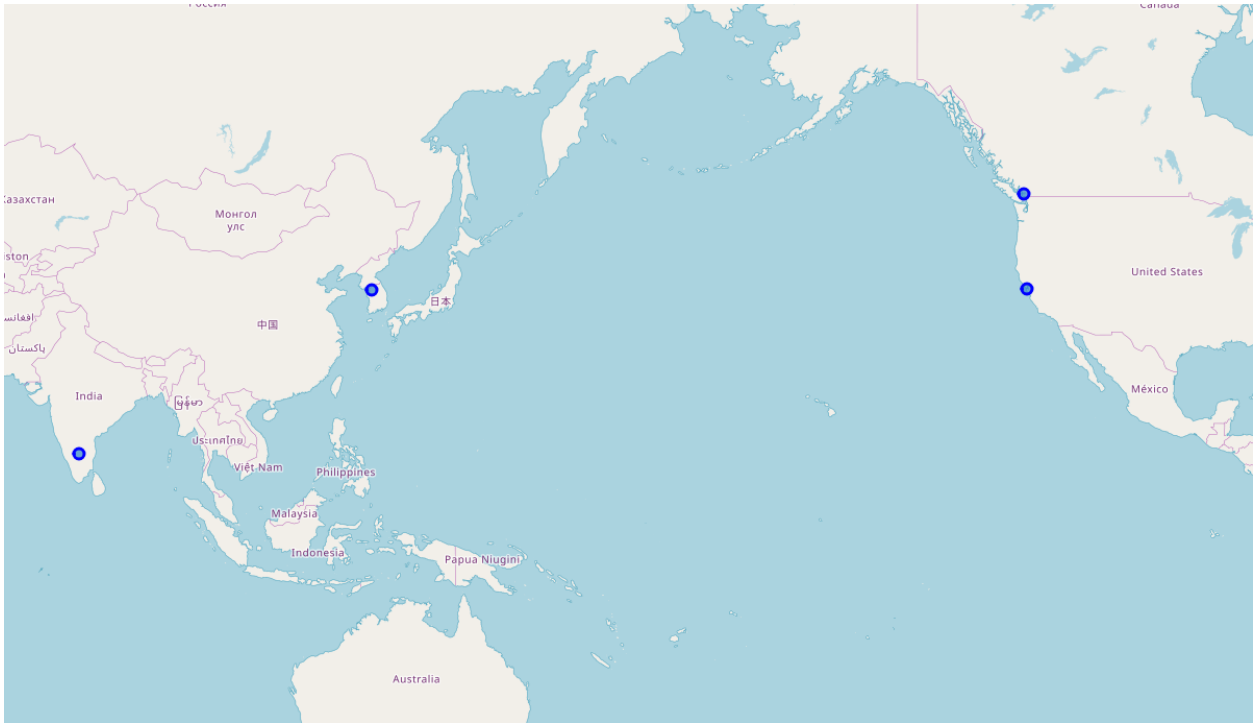
Even after providing a 2 sec gap in calls, there are timeout errors. So, to handle these errors is simple. Simply call the Nominatim service again for these locations after checking for network connectivity.

Missing/No Coordinates

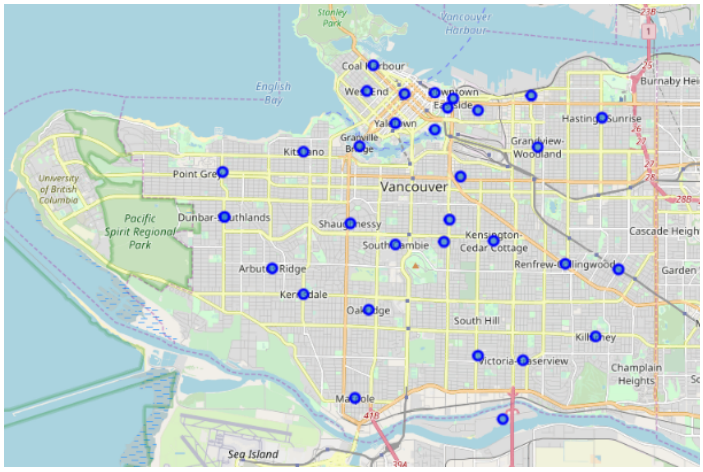
Some locations will not resolve into coordinates. This can happen because some locations may have different spellings. These can be rectified by using different spellings. Some locations will not resolve despite that. Then that data is procured manually searching on Google Maps.

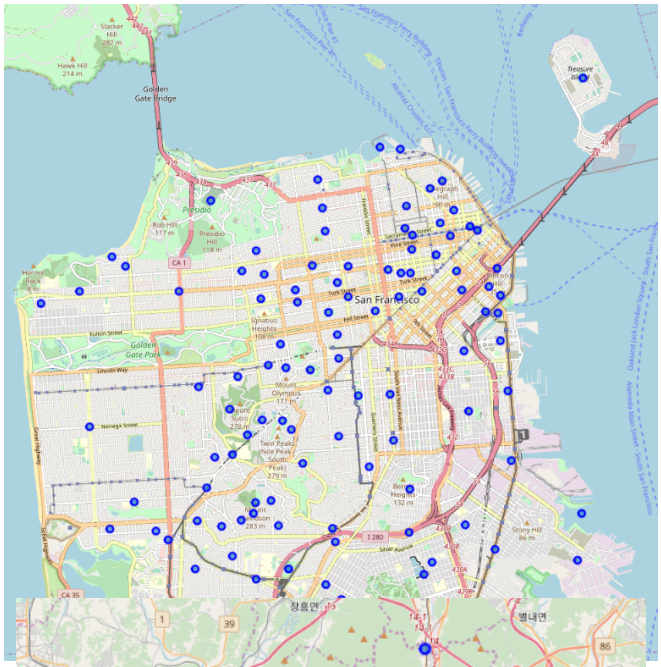
Maps

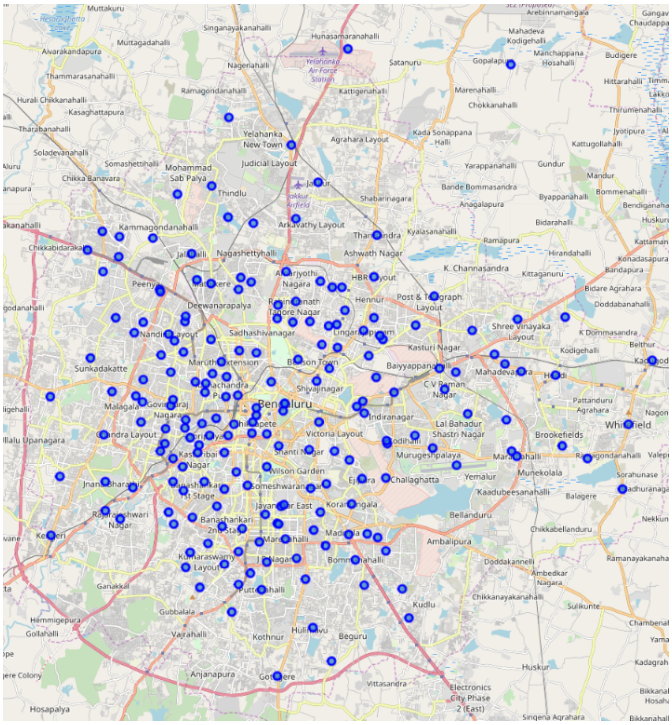
Maps are generated for each city with neighbourhoods shown as markers. But before that a world map is created with the cites as markers. Below is the world map.

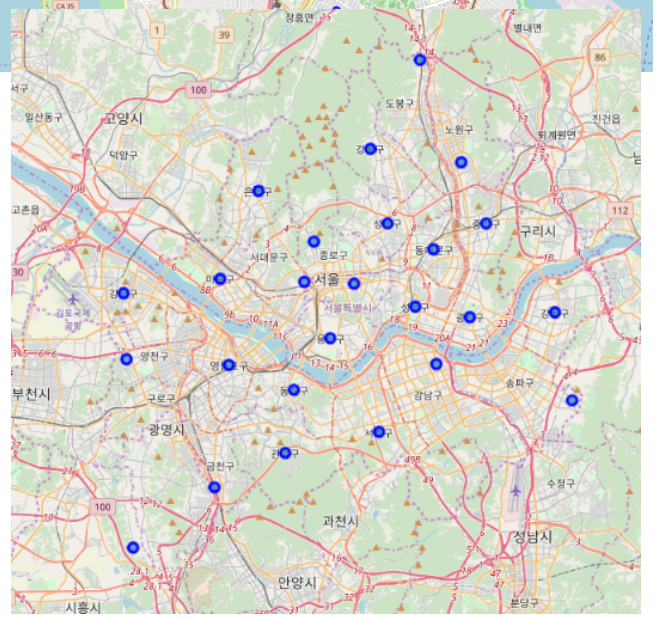


Maps of the cities are below.









Venues

Getting Venues

Venues are places located in the neighbourhoods like restaurants, hotels, cafes, parks etc. Foursquare API was used get the list of venues for a neighbourhood. Since free version of the Foursquare API is used, a maximum of 100 venues can be retrieved.

Now the size of each neighbourhoods are not equal. Especially between cities. To search venues about position of the coordinates, radius needs to be given. So, for Bengaluru and Vancouver the radius considered is 1000m. For Seoul, radius is 2500m; for Vancouver and San Francisco, radii are 500m.

City	Area (km^2)	No. of Neighbourhoods	Avg Neighbourhood Radius Considered (m)
Bengaluru	709	195	1000
Seoul	605.2	25	2500
Vancouver	115	33	500
San Francisco	121.4	114	500

The venues for each location are stored in separate dataframes.

Studying Venues

To study the venues, the dataframes containing the venues are grouped by neighbourhoods and summed up.

For Bengaluru, there are some neighbourhoods with a lot of venues while many have very few venues. Realistically this is not true, and this can be considered as Foursquare not having detailed venues for all neighbourhoods. My assumption is that venues in Foursquare are more recognizable and an international restaurant franchisee will like to be in neighourhoods with more recognizable venues.

For individual clustering, one hot encoding is done for neighbourhoods of each location. While for complete clustering, all venues are combined into the same dataframe and then one hot encoding is done. This makes sure that all the types of venues are considered.

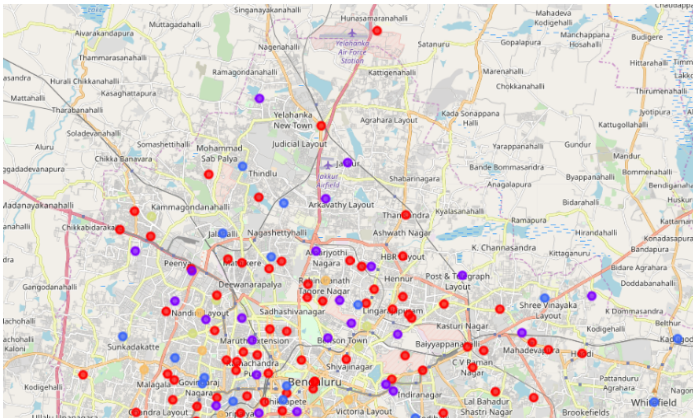
Individual Clustering Results

Individual Clustering will help understand how the individual locations can be clustered. To be consistent with all the individual location clustering and the complete clustering, there are going to be 8 clusters.

It must be noted that cluster labels are not the same across different locations. So, Cluster 0 in Bengaluru is not the same as Cluster 0 in San Francisco.

Bengaluru

Looking at the clustering of the neighbourhoods in Bengaluru, there are 5 clusters with 3 possible outliers, with majority of the neighbourhoods in cluster 0. The neighbourhoods of all



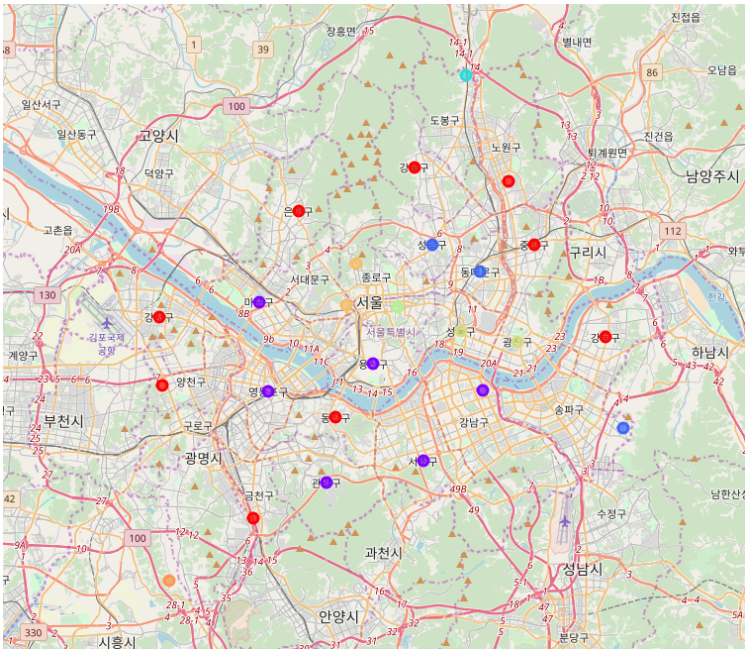
clusters look equally distributed on the map.

Cluster Labels	Neighbourhood
0	108
1	9
2	1
3	1
4	45
5	10
6	1
7	20

Seoul

Cluster Labels	Neighbourhood
0	9
1	2
2	2
3	3
4	1
5	6
6	1
7	1

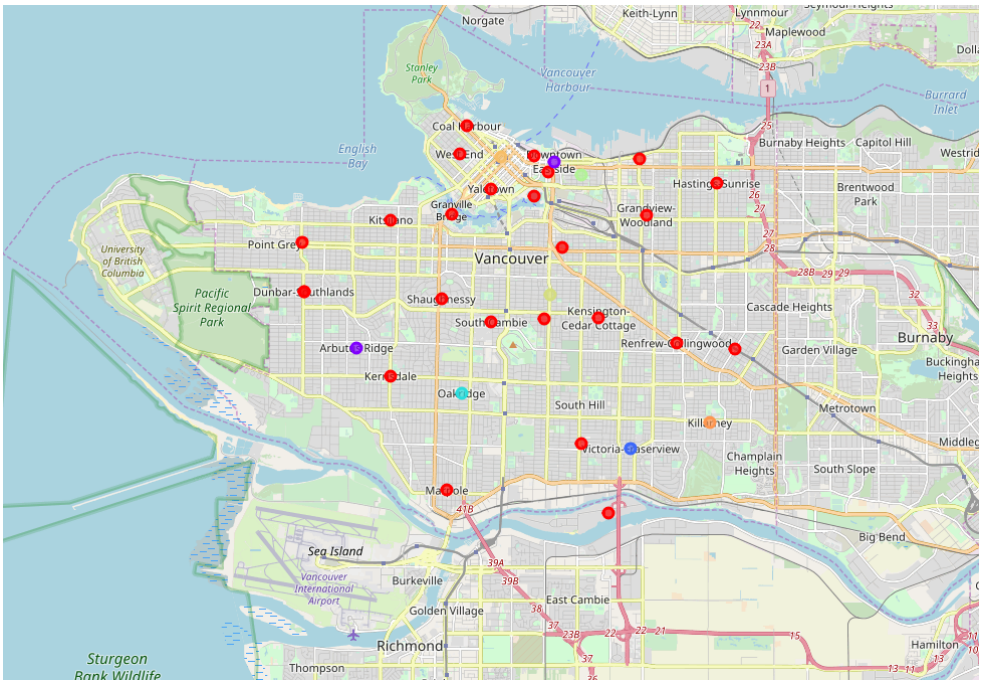
In



Seoul, there are only 25 neighbourhoods, so it's difficult to tell if the clusters with only one neighbourhood are outliers or not. Looking at map, neighbourhoods in cluster 0 are at the edges of the city.

Vancouver

There are 33 neighbourhoods, so it's difficult to tell if the clusters with only one neighbourhood are outliers or not. Most of the neighbourhoods are in cluster 1.



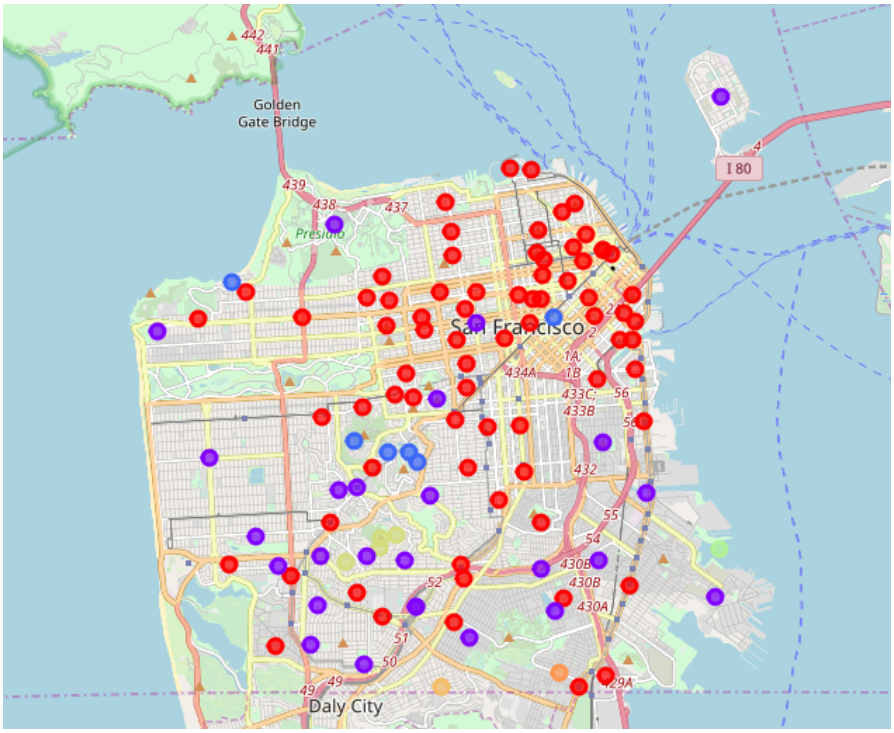
Neighbourhood	
Cluster Labels	
0	1
1	25
2	1
3	1
4	2
5	1
6	1
7	1

San Francisco

Neighbourhood	
Cluster Labels	
0	1
1	1

The neighbourhoods can be divided into 4 clusters and possible 4 possible outliers.

Cluster Labels	City	Neighbourhood
0	San Francisco	1
1	Bengaluru	12
	San Francisco	6
2	Bengaluru	15
	San Francisco	6
	Seoul	1
	Vancouver	2
3	Bengaluru	1
4	Bengaluru	1
5	Bengaluru	64
	San Francisco	35
	Seoul	15
	Vancouver	17
6	Bengaluru	102
	San Francisco	65
	Seoul	9
	Vancouver	14
7	San Francisco	1



Complete Clustering Results

The complete clustering gives some interesting results. Seoul and Vancouver can be divided into three clusters (2, 5, 6) which the other locations also have. Bengaluru and San Francisco have a lot of common clusters (1, 2, 5, 6). There are some possible outliers in San Francisco (0, 7) and Bengaluru (3, 4) which don't fit in any common clusters.

The big takeaway from this is that there are three clusters with Neighbourhoods from all the locations (2, 5, 6). So, these Neighbourhoods can be considered similar based on the venues present in them.

Discussion

The objective of this analysis was that if there is a restaurant franchise in both Vancouver and San Francisco and we want to open a new franchise in Bengaluru and Seoul then in which Neighbourhoods of the cities they should open. Based on Complete Clustering Neighbourhoods in clusters 2, 5 and 6 are similar Neighbourhoods. So, if the restaurant in the Neighbourhoods of these clusters in Vancouver and San Francisco then a new franchise can be opened in the Neighbourhoods of the same clusters in Bengaluru and Seoul. Since majority of the Neighbourhoods of all the locations are in these three clusters then there is a good probability of finding a match.

Conclusion

The result showed that the restaurant franchise can be opened in Bengaluru and Seoul though more data and analysis is needed. More data like the customer rating and pricing details will help but with the free Foursquare API there is limited access to the required data.