Predicting a suitable Neighboorhood for a Hotel using Foursquare data

Elie Chdid
October 2020

Introduction

My client owns a successful Hotel in "Stuyvesant Town" in Manhattan, NY and now has the opportunity to gow their business and open a second Hotel in the city of Düsseldorf in Germany. Often the success of a Hotel is linked to its location since it is one of the most important criteria for most customers staying at a Hotel. Given the success of the Hotel in Manhattan, my client would like to open the 2nd Hotel in Düsseldorf City Center. Before doing any further in depth analysis on any neighborhoods in Düsseldorf, my client would like to narrow down their options by focusing on neighborhoods that are comparable to "Stuyvesant Town" using data.

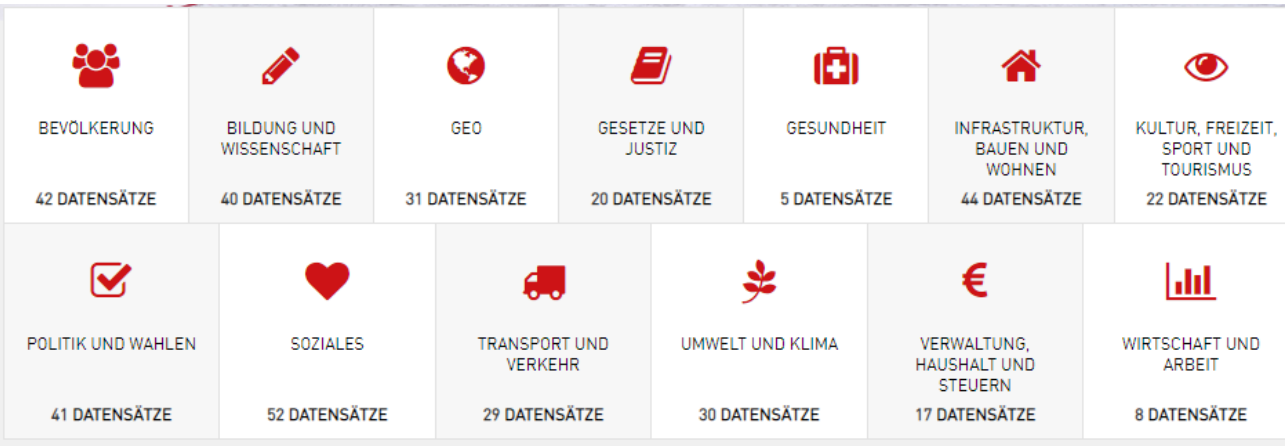
My task is to use and analyze the available online data to give location recommendations to my client based on their requirements.

The Data

There is plenty of data available online but we will be using the official data povided by the city's official website "https://opendata.duesseldorf.de/" for location and neighborhood information. We will also be exploring the the venues in target neighborhoods using the Foursquare services. The client has also provided a reference venue data for "Stuyvesant Town" on which the anaysis should be based.

Düsseldorf City Data:

As you can see on the below screenshot from https://opendata.duesseldorf.de/ the city provides a number of Data that is open and available for every one. You can choose the topic of interest and then select the file type that you would like to work with.



For our purpose we will be using the data file „Stadtteile Düsseldorf 2017“ in .csv format. Stadtteile

is the german word for Neighborhoods and using the built-in preview option on the website we could quickly verify if the data has the necessary information. In the Data we will find the name of the Neighborhood „Stadtteil“, the Neighborhood's number Id „Stadtteilnummer“ and the District number Ids „Stadtbezirksnummer“

stadtteil	stadtteilnummer	stadtbezirksnummer
Carlstadt	012	1
Derendorf	015	1
Düsseltal	023	2
Eller	082	8
Flehe	038	3
Flingern Nord	022	2
Flingern Süd	021	2
Friedrichstadt	031	3
Garath	101	10

Foursquare Data:

The Foursquare data will be in .json file formate and when visualized it looks a bit messy. Howerver when undersstood and wrangled properly in could be transformed into a comprehensive dataframe containing all the information we need about venues in any given neighborhood.

Example Raw .json Data


```
{'type': 'FeatureCollection',
 'totalFeatures': 306,
 'features': [{'type': 'Feature',
   'id': 'nyu_2451_34572.1',
   'geometry': {'type': 'Point',
    'coordinates': [-73.84720052054902, 40.894705176
   'geometry_name': 'geom',
   'properties': {'name': 'Wakefield',
    'stacked': 1,
    'annoline1': 'Wakefield',
    'annoline2': None,
    'annoline3': None,
    'annoangle': 0.0,
    'borough': 'Bronx',
    'bbox': [-73.84720052054902,
     40.89470517661,
     -73.84720052054902,
     40.89470517661]}},
  {'type': 'Feature',
```


Example Data after wrangling

	name	categories	lat	lng
0	Bikram Yoga	Yoga Studio	40.876844	-73.906204
1	Arturo's	Pizza Place	40.874412	-73.910271
2	Tibbett Diner	Diner	40.880404	-73.908937
3	Starbucks	Coffee Shop	40.877531	-73.905582
4	Dunkin'	Donut Shop	40.877136	-73.906666

Client reference Data:

The most frequent venues in Stuyvesant Town are as following:

1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Park	Harbor / Marina	Cocktail Bar	Baseball Field	Bar

6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
Gym / Fitness Center	Heliport	Bistro	Farmers Market	Boat or Ferry

Methodology

The task given by the client is very clear. Recommend a potential neighborhood based on the data reference that was given. The reference data outlined the 10 most frequent venues in „Stuyvesant Town“ and we need to gather the same data for neighborhoods in Düssedorf in order to make a comparison with the reference data and make subsquent recommendation.

We will first explore the data at hand then wrangle it into a format that we can work. Comparing each neighborhood with „Stuyvesant Town“ could be difficult and very time consuming so after we wrangle the data into the requiered format we will run a K-means algorithm on the data and cluster them into clusters based on the frequency of venues in each neighborhood. This will allow us to compare the reference data to clusters of similar neighborhoods instead of comparing it to each neighborhood on its own.

1- Data Exploration and Wrangling

The raw data downloaded from the Düsseldorf city website and the data acquired through Forusquare cannot be used without further exploration and wrangling.

1.1 - The Düsseldorf City Data

Here is a quick look at the raw data

	Stadtteil;Stadtteilnummer;Stadtbezirksnummer
0	Altstadt;011;1
1	Angermund;055;5
2	Benrath;095;9
3	Bilk;036;3
4	Carlstadt;012;1

As we can see, it needs a lot of wrangling to be useful for our purpose. First we need to split the values and attributes seperated by a „ ; “ into their own separate columns so that we have 3 additional columns „Stadtteil“, „Stadtteilnummer“ and „Stadtbezirksnummer“. Which will make our dataframe look as following

	Stadtteil;Stadtteilnummer;Stadtbezirksnummer	Stadtteil	Stadtteilnummer	Stadtbezirksnummer
0	Altstadt;011;1	Altstadt	011	1
1	Angermund;055;5	Angermund	055	5
2	Benrath;095;9	Benrath	095	9
3	Bilk;036;3	Bilk	036	3
4	Carlstadt;012;1	Carlstadt	012	1

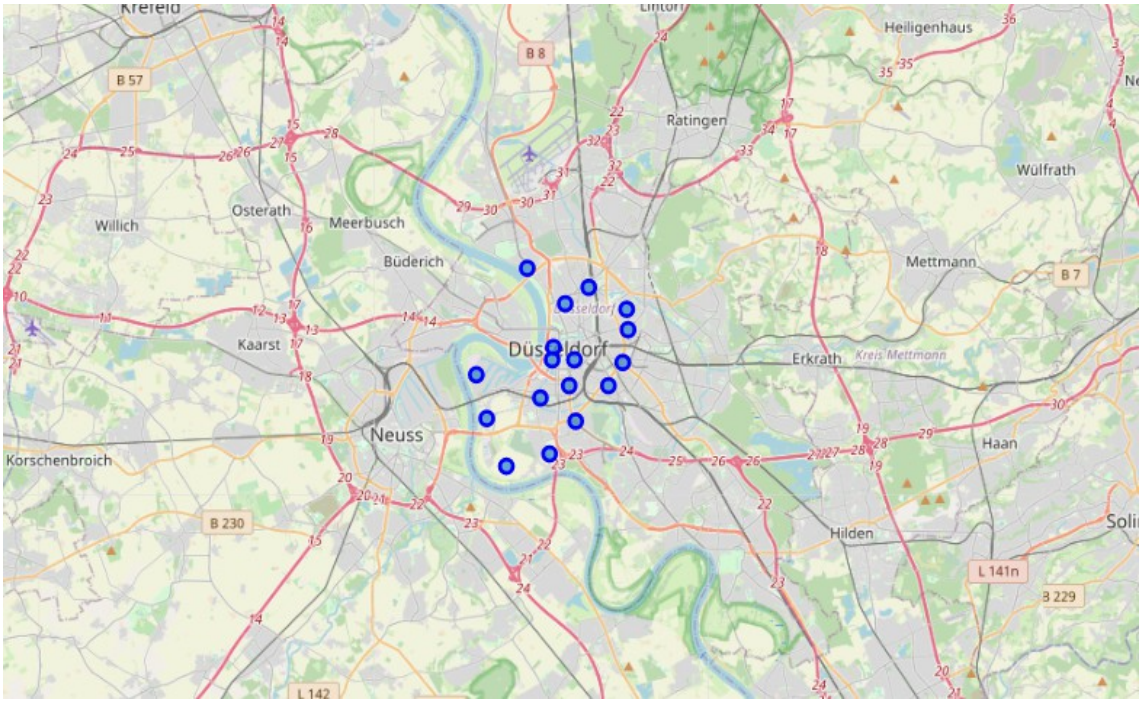
Next to narrow down our target neighborhoods based on the client's requirement. At the moment all the neighborhoods (Stadtteil) in Düsseldorf are included but we want to only analyze the neighborhoods that are in the city center. After a quick research, it is evident that Düsseldorf City Center is made up of District 1, 2 and 3 (Stadtbezirksnummer) which means we need to get rid of all of the rows/neighborhoods that are not in Districts 1,2 or 3.

After that we will get rid of the uncessary columns like „Stadtteilnummer“, „Sadtbezirksnummer“ and „Stadtteil;Stadtteilnummer;Stadtbezirksnummer“. And rename the column „Stadtteil“ to „Neighborhood“. Resulting in a data frame that contains only one column titled „Neighborhood“ and contains all the neighborhoods that are in the city center of Düsseldorf which we will call target neighborhoods.

Later on our aim is to combine this dataframe with the data acquired through Foursquare and visualize the neighborhoods on the map which is why we need to add the coordinates of each neighborhood to the table. Luckily this is easily achieved by running a Python „for loop“ that uses the geolocator function. This will give us a list of the coordinates which we could append to our dataframe resulting in a table that has all the target neighborhoods together with their coordinates. Let's call it „df_duss_coor“.

	Neighborhood	latitude	longitude
0	Altstadt	51.225912	6.773567
1	Bilk	51.202758	6.785101
2	Carlstadt	51.222142	6.773394
3	Derendorf	51.244549	6.792249
4	Düsseltal	51.237841	6.812116
5	Flehe	51.192204	6.771713
6	Flingern Nord	51.231381	6.813238
7	Flingern Süd	51.221009	6.810060
8	Friedrichstadt	51.213564	6.781700
9	Golzheim	51.250794	6.759963
10	Hafen	51.217029	6.733576
11	Hamm	51.203572	6.738809
12	Oberbilk	51.213689	6.802428
13	Pempelfort	51.239601	6.779685
14	Stadtmitte	51.221939	6.784423
15	Unterbilk	51.210055	6.766965
16	Volmerswerth	51.188578	6.749010

Now let us get ourserlves more familiar with the neighborhoods by plotting them on a map using the folium library.



We can see that all the neighborhoods are indeed all centered around Düsseldorf city center and none of the outskirt neighborhoods are considered. Knowing that we are on the right track we can now proceed to exploring those neighborhoods using Foursquare.

1.2 - The Foursquare Data

The raw Foursquare data comes in the form of a „.json“ file which upon inspection looks like the following figure.


```
{'meta': {'code': 200, 'requestId': '5f96b0c908bd407f2d4310d1'},
 'response': {'suggestedFilters': {'header': 'Tap to show:',
   'filters': [{'name': 'Open now', 'key': 'openNow'}]},
  'headerLocation': 'Altstadt',
  'headerFullLocation': 'Altstadt, Düsseldorf',
  'headerLocationGranularity': 'neighborhood',
  'totalResults': 188,
  'suggestedBounds': {'ne': {'lat': 51.2299018045, 'lng': 6.783485825221867},
   'sw': {'lat': 51.2209017955, 'lng': 6.769141574778134}},
  'groups': [{'type': 'Recommended Places',
    'name': 'recommended',
    'items': [{'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '56eb2d9bcd10c4efdfb581cf',
       'name': 'Casita Mexicana',
       'location': {'address': 'Hunsrückenstr. 15',
        'lat': 51.22667595657295,
        'lng': 6.775478313848207,
        'labeledLatLngs': [{'label': 'display',
          'lat': 51.22667595657295,
```

Obviously the data in this form is not useful to us and we need to deploy some data wrangling technique to import all the useful data to our dataframe „df_duss_coor“.

Looking at the data in the .json file we can see that all the information we need is in the "items" key. Like venue name, latitude, longitude, category type. We will now define a function that will return all that information into our "df_duss_coor" dataframe

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Altstadt	51.225912	6.773567	Casita Mexicana	51.226676	6.775478	Mexican Restaurant
1	Altstadt	51.225912	6.773567	Rösterei VIER	51.224536	6.773703	Coffee Shop
2	Altstadt	51.225912	6.773567	Rösterei VIER	51.225940	6.772294	Coffee Shop
3	Altstadt	51.225912	6.773567	Elephant Bar	51.226851	6.772636	Cocktail Bar
4	Altstadt	51.225912	6.773567	Bar Chérie	51.226886	6.772424	Bar

The resulting dataframe has 668 rows which is the total number of venues that are located in the target neighborhoods. Of those 668 venues there are 152 unique venue categories.

Now that our data is ready we can proceed to the data analysis.

2- Data Analysis

Now that we have put the data into the required dataframe format we can begin analysing the data. Let's see how the neighborhoods compare when it comes to abundance of venues

	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
Altstadt	100	100	100	100	100	100
Bilk	20	20	20	20	20	20
Carlstadt	100	100	100	100	100	100
Derendorf	24	24	24	24	24	24
Flehe	8	8	8	8	8	8
Flingern Nord	37	37	37	37	37	37
Flingern Süd	28	28	28	28	28	28
Friedrichstadt	87	87	87	87	87	87
Golzheim	14	14	14	14	14	14
Hafen	3	3	3	3	3	3
Hamm	11	11	11	11	11	11
Oberbilk	22	22	22	22	22	22
Pempelfort	69	69	69	69	69	69
Stadtmitte	100	100	100	100	100	100
Unterbilk	45	45	45	45	45	45

The aim of this analysis is to be able to identify a suitable location for the Hotel based on the reference data provided by the client. The reference data is a table listing the 10 most popular venues in "Stuyvesant Town". To be able to find a similar neigborhood in Düsseldorf we need to cluster and classify the target neighborhoods based on the 10 most poplular venues for each neighborhood. In order to do that we need to turn all the venues categories into separate attributes and calculate the frequency of each category per neighborhood. For that we can use „one hot encoding method“ which will return a table with venue categories as attributes and a "0" or "1" as values. With "0" meaning no such venue existst in the neighborhood and "1" meaning such venue exists.

	Neighborhood	American Restaurant	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Automotive Shop	BBQ Joint	Baby Store	Bakery	Bank	Bar	Beach	Beer Bar	B Garc
0	Altstadt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	Altstadt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	Altstadt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	Altstadt	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	Altstadt	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

The new dataframe "duss_onehot" has 153 columns which validates our code since one column is the neighborhoods name and the remaining 152 columns represent the 152 unique categories found in the target neighborhoods. Now let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category. This will help is later in identfying the most frequent venues per neighborhood. The higher the frequency of a category the more frequently it exists in the given neighborhood.

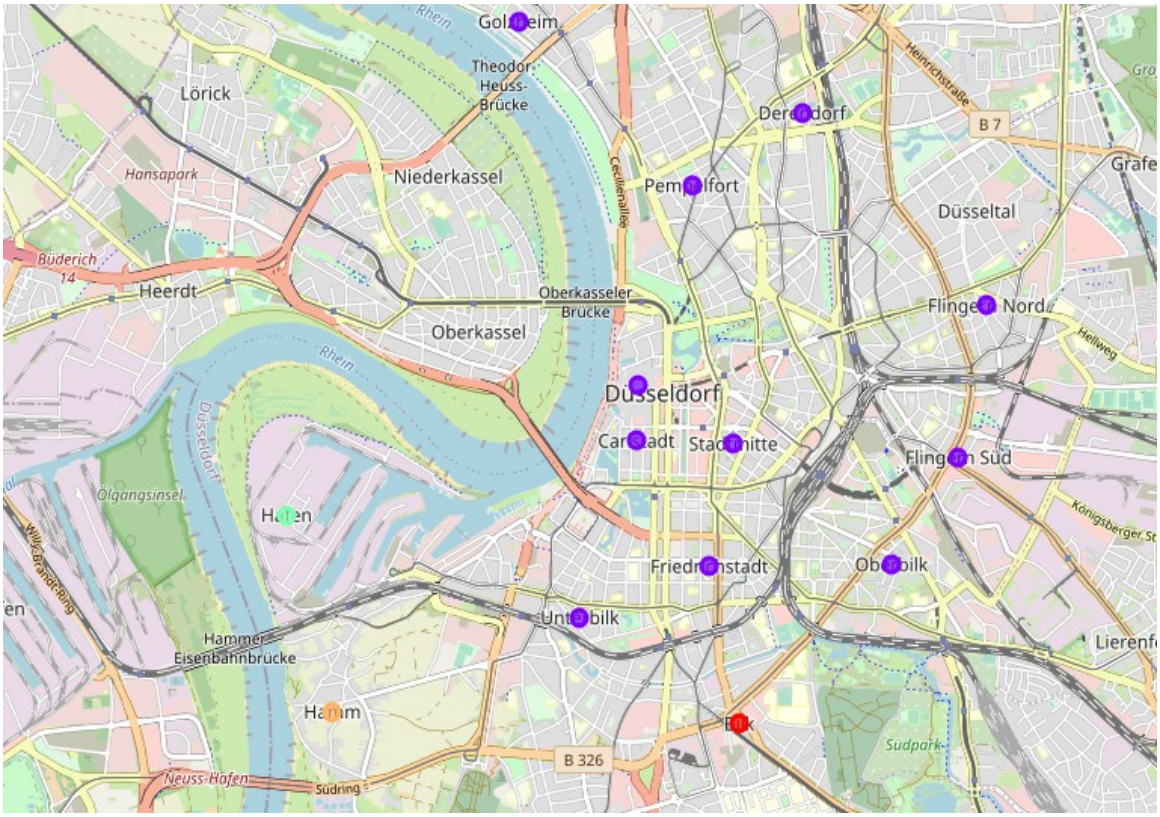
After calulating the frequency of occurence for each venue in each neighborhood we can now run a function that will return the 10 most frequent venues in each neighborhood which will result in a dataframe that is matching with the reference data given by the client. Let's call this dataframe „neighborhoods_venues_sorted“.

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Altstadt	Café	Plaza	Brewery	Coffee Shop	Bar	Italian Restaurant	Steakhouse	Boutique	Ice Cream Shop	German Restaurant
1	Bilk	Tram Station	Hotel	Bakery	Greek Restaurant	Doner Restaurant	Shipping Store	Costume Shop	Café	Supermarket	Italian Restaurant
2	Carlstadt	Italian Restaurant	Coffee Shop	Boutique	Café	Clothing Store	Plaza	Ice Cream Shop	German Restaurant	Bakery	Brewery
3	Derendorf	Liquor Store	Supermarket	Restaurant	Colombian Restaurant	Shipping Store	Gym / Fitness Center	Juice Bar	Kids Store	German Restaurant	Gastropub
4	Flehe	Soccer Field	Skate Park	Tram Station	Gym	Park	Bakery	Drugstore	Doner Restaurant	Electronics Store	Dive Bar

3- Modeling and Clustering the Data

Now that the Data has been properly structured we can choose to compare line by line and each neighborhood to our data reference and give recommendations. This is only possible because we only have 16 target neighborhoods but it is tedious work and could get really confusing. Not to mention it would be impossible to do it manually in case we had a larger number of neighborhoods. To solve this problem we will choose to cluster the neighborhoods in different clusters depending on the similarity in venue frequency between the neighborhoods. To do that we will use k-means to cluster the neighborhoods into 5 different clusters and then we will recommend the cluster that best matches our reference data.

After running the K-means algorithm we visualize the result on the map. The 5 different clusters are labled by the colors purple, green, orange, red and blue. It is reassuring that there is also a geographical difference between the clusters which is common when clustering neighborhoods.



4- Examining and discussing the results

Cluster 0

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
1	Bilk	Tram Station	Hotel	Bakery	Greek Restaurant	Doner Restaurant	Shipping Store	Costume Shop	Café	Supermarket	Italian Restaurant

Cluster 1

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Altstadt	Café	Plaza	Brewery	Coffee Shop	Bar	Italian Restaurant	Steakhouse	Boutique	Ice Cream Shop	German Restaurant
2	Carlstadt	Italian Restaurant	Coffee Shop	Boutique	Café	Clothing Store	Plaza	Ice Cream Shop	German Restaurant	Bakery	Brewery
3	Derendorf	Liquor Store	Supermarket	Restaurant	Colombian Restaurant	Shipping Store	Gym / Fitness Center	Juice Bar	Kids Store	German Restaurant	Gastropub
5	Flingern Nord	Café	Asian Restaurant	Bakery	Pizza Place	Greek Restaurant	Italian Restaurant	Vietnamese Restaurant	Hotel	German Restaurant	Office
6	Flingern Süd	Hotel	Portuguese Restaurant	Electronics Store	Greek Restaurant	Italian Restaurant	Gourmet Shop	Doner Restaurant	Music Venue	Rock Club	Chinese Restaurant
7	Friedrichstadt	Hotel	Café	Vietnamese Restaurant	Bakery	Pizza Place	Bar	Italian Restaurant	Pub	Miscellaneous Shop	Middle Eastern Restaurant
8	Golzheim	Italian Restaurant	Bakery	Café	Metro Station	Salad Place	Fast Food Restaurant	Steakhouse	Supermarket	Beach	Modern European Restaurant
11	Oberbilk	Hotel	Bar	Hookah Bar	Korean Restaurant	Massage Studio	Metro Station	Mobile Phone Shop	Drugstore	Doner Restaurant	Comfort Food Restaurant
12	Pempelfort	Italian Restaurant	Café	Ice Cream Shop	Bakery	Hotel	Drugstore	Supermarket	Restaurant	Trattoria/Osteria	Vietnamese Restaurant
13	Stadtmitte	Japanese Restaurant	Korean Restaurant	Café	Hotel	Ramen Restaurant	Italian Restaurant	Coffee Shop	Grocery Store	Shopping Mall	Sushi Restaurant
14	Unterbilk	Café	Italian Restaurant	Restaurant	Ice Cream Shop	Pizza Place	Coffee Shop	Deli / Bodega	Dive Bar	Pub	Plaza

Cluster 2

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
4	Flehe	Soccer Field	Skate Park	Tram Station	Gym	Park	Bakery	Drugstore	Doner Restaurant	Electronics Store	Dive Bar

Cluster 3

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
9	Hafen	Scenic Lookout	Boat or Ferry	River	Yoga Studio	Fried Chicken Joint	French Restaurant	Fountain	Food	Fast Food Restaurant	Farmers Market

Cluster 4

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
10	Hamm	Beach	Café	Gastropub	Garden	German Restaurant	Greek Restaurant	Bank	Hotel	Restaurant	Drugstore

From the results we can see that most of the neighborhoods in Düsseldorf City Center are put together in cluster 1 which is clearly the touristic part of Düsseldorf which has all the restaurants, bars, clubs, cafes and hotels. The neighborhoods on the southern outskirts of Düsseldorf City Center were all put in their own clusters based on their attributes. We could tell that cluster 0 or neighborhood „Bilk“ is a residential neighborhood with supermarkets, bakeries and tram stations.

Cluster 2 or neighborhood „Flehe“ looks also to be residential but more family friendly with parks, soccer fields, skate parks which makes sense geographically given that cluster 2 is closer to the outskirt of the City Center. Clusters 3 and 4 seem to be similar which is probably driven by their proximity to the river.

5- Conclusion

Client Reference Data

1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Park	Harbor / Marina	Cocktail Bar	Baseball Field	Bar

6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
Gym / Fitness Center	Heliport	Bistro	Farmers Market	Boat or Ferry

Looking at the reference Data we can quickly dismiss cluster 1 which happens to be the largest cluster with the most venues. It is packed with restaurants, cafes and bars which is not the kind of neighborhood we are looking for.

Comparing the reference data to the remaining clusters we can recommend the most suitable location to be in Cluster 3 which is neighborhood "Hafen" which means „harbor“ in german. Stuyvesant and Hafen are similar neighborhoods since they are both on the river side, have harbors/boat/ferries, they contain public recreational venues like Fountains and Parks and they both have a farmer's market as one of the most frequent enue categories. In addition to the similarities it is worth noting that "Hafen" does not have hotels which minimizes potential competition and has few restaurants which is beneficial for the hotel's restaurant.

The second choice would be Cluster 4 or neighborhood "Hamm" which is adjacent to "Hafen" and has similar attributes.