Recommending the best-matching neighborhoods

Carlos Ruiz
November 12, 2019

1. Introduction

1.1. Background

When somebody needs relocation to a new city, a common concern is how to choose the right neighborhood to live.

Of course, pricing is always a key parameter, as much for buying as for renting. But the surroundings are also important, taking in consideration what kind of places are available in the nearby (like restaurants, shops, cinemas, ...), but also other facilities like transportation, parks, schools, libraries, hospitals, etc.

1.2. Problem

The appropriate neighborhood could be different depending on people habits and tastes. In fact, every person has specific needs, depending on age, hobbies, family, workplace, so the 'best' neighborhood could be really different. It would be always the place which matches as close as possible to his/her lifestyle.

The data problem is how to select or recommend a specific area for somebody who is moving to a new city, according to some personal preferences or needs, which is actually a realistic scenario for a real estate business.

1.3. Interest

Being able to recommend the best neighborhood for relocating people could be very beneficial for them, improving their experience in the long term.

Often, there is no previous knowledge about the destination city. Choosing can require some documentation effort and, without references, may be difficult and error prone. And in many situations, they must move immediately, so timing is also important.

The recommendation service could be really advantageous for real estate business or relocation helpers, as an added-value capability that may have a direct effect on customer satisfaction.

2. Data acquisition and cleaning

2.1. Data sources

The main data source to classify the neighborhoods limits should be the administrative division, for any geographical area in analysis, and this information will require a specific data source for every location.

Within the described procedure, the New York city is used as example, and to ensure complete availability, the known and previously used 'New York City Neighborhoods' file is the chosen one. This information is offered from the New York (City) Department of City Planning, and can be retrieved from the NYU Spatial data Repository.

Also from the Department of City Planning, a 'City Planning Facilities Database' can be useful. The file can be downloaded for the web page 'BYTES of the BIG APPLE', and includes many public facilities, already aligned with the previous administrative division, according to the following main categories:

- Health and Human Services
- Education, Child Welfare, and Youth
- Parks, Gardens, and Historical Sites
- Libraries and Cultural Programs
- Public Safety, Emergency Services, and Administration of Justice
- Core Infrastructure and Transportation
- Administration of Government

However, most of the listed categories are also available in the Foursquare classification. Apart of including also leisure venues, using Foursquare API instead of a static list could help us to filter data to include only working and/or trending places or even to use also the customer tastes. In any of the cases, the city facilities database will be used only if enrichment is required for a specific, missing type of service.

The lookup to create the base dataframe will be performed by calling the search Foursquare regular API endpoints for every one of the New York neighborhoods. Existing venues and their categories will be a mean to rank the neighborhood according to its closeness to the customer likings.

So, a second information source requirement comes from the need to classify these different venue categories, according to some preferences profiles. The mapping could be gathered by specific surveys, tied to a previous market segmentation in different groups.

Examples of the segmentation characteristics could be age (using several ranges), marital status, children (young or teenagers), restaurant or cinema lovers, having a dog, need of public transportation, etc. In fact, any attribute leading to groups that will need or like diverse facilities in their living area.

And a third information source is a profile for the target customer. It is just a compilation of characteristics that could have an impact needs or tastes, gathered by direct request, through interview or application form. They should be the same segmentation characteristics used to classify the venue categories, to allow aligned comparison with them.

2.2. Data cleaning

The New York data format is a JSON file, from which only the features key is retrieved, containing a list of neighborhoods. The data structure for each of them includes identification of the parent borough and geographical coordinates.


```
{'type': 'Feature',
 'id': 'nyu_2451_34572.1',
 'geometry': {'type': 'Point',
  'coordinates': [-73.84720052054902, 40.89470517661]},
 '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]}}
```


There is no need to clean this information, which includes 306 neighborhoods within 5 boroughs.

```
neighborhoods.head()
```

	Borough	Neighborhood	Latitude	Longitude
0	Bronx	Wakefield	40.894705	-73.847201
1	Bronx	Co-op City	40.874294	-73.829939
2	Bronx	Eastchester	40.887556	-73.827806
3	Bronx	Fieldston	40.895437	-73.905643
4	Bronx	Riverdale	40.890834	-73.912585

The Foursquare information about venues provides a list of categorized venues with its location coordinates. As the searching area must be limited, a radius parameter (from the neighborhood coordinates) is fixed to 500 m.

As well, the number of retrieved venues is limited to a parameter value of 100. Apart of avoiding a huge number of items, it limits the effect of some tourist or downtown neighborhoods having a very high number of venues of specific categories, which add an undesired bias in the final scoring.

The Foursquare search service is called for each of the listed neighborhoods, resulting in a total of 10258 selected venues

```
print(ny_venues.shape)
ny_venues.head()
```

```
(10258, 7)
```

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	\
0	Wakefield	40.894705	-73.847201	
1	Wakefield	40.894705	-73.847201	
2	Wakefield	40.894705	-73.847201	
3	Wakefield	40.894705	-73.847201	
4	Wakefield	40.894705	-73.847201	

	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Lollipops Gelato	40.894123	-73.845892	Dessert Shop
1	Rite Aid	40.896649	-73.844846	Pharmacy
2	Carvel Ice Cream	40.890487	-73.848568	Ice Cream Shop
3	Shell	40.894187	-73.845862	Gas Station
4	Dunkin'	40.890459	-73.849089	Donut Shop

However, Foursquare is providing venues for all the existing categories in each of the analyzed areas, while we are interested in selecting only the most significant categories for every neighborhood. That is, avoiding outliers, which mean categories being represented by a very small number of venues.

After adding a hot encoding for venues and grouping by category, the full list is reduced to just the 10 most representative categories, ordering by the total number of venues. These 10 categories and the number of venues for each of them is a kind of signature or profile for the neighborhood.

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	\
0	Allerton	Pizza Place	Deli / Bodega	
1	Annadale	Pizza Place	Liquor Store	
2	Arden Heights	Pizza Place	Rental Car Location	
3	Arlington	Intersection	American Restaurant	
4	Arrochar	Bus Stop	Deli / Bodega	

	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	\
0	Chinese Restaurant	Supermarket	Bakery	
1	American Restaurant	Pharmacy	Park	
2	Pharmacy	Coffee Shop	Women's Store	
3	Bus Stop	Grocery Store	Filipino Restaurant	
4	Italian Restaurant	Pizza Place	Food Truck	

	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	\
0	Fast Food Restaurant	Electronics Store	Pharmacy	
1	Diner	Restaurant	Food	
2	Ethiopian Restaurant	Event Service	Event Space	
3	Event Space	Exhibit	Eye Doctor	
4	Liquor Store	Sandwich Place	Athletics & Sports	

	9th Most Common Venue	10th Most Common Venue
0	Martial Arts Dojo	Grocery Store
1	Sports Bar	Dance Studio
2	Exhibit	Eye Doctor
3	Factory	Falafel Restaurant
4	Bagel Shop	Supermarket

2.3. Feature selection

From the New York administrative information, we are fetching the name of the neighborhoods and their geographical coordinates. The name of the borough, as a parent entity, is also included, but it is not used for the later analysis.

From Foursquare, we are fetching the venue names, the location (also in the form of coordinates) and, very important, the category. The category will be used in the process for grouping, to select the most relevant ones in the area.

The profile for a customer is made from a list of yes/no characteristics, selected attributes that affect the venue categories that will match best with his/her preferences. The list can be altered and extended, but for the current implementation the following characteristics have been chosen:

- **Young**: Juniors, people under 30. May be studying.
- **Mature**: Middle aged, between 30 and 55. Usually working.
- **Old**: Seniors, people over 55. May have grandsons.
- **Single**: Not married, living alone.
- **Married**: Married, living with a couple. Shared activities
- **Children**: Having kids. Needing facilities for them
- **Pets**: Animals, requiring a place to walk them
- **Music**: Loving places to dance or live concerts
- **Sports**: Asking for sports facilities, for training or watching matches
- **Culture**: Looking for theaters, museums, libraries, …
- **Fashion**: Loves shopping, malls, hairdressing
- **Going-out**: Social life, bars, pubs, clubs
- **Health**: Likes parks, walking, spas and saunas,
- **Religious**: Needing churches, synagogues, mosques

All the venue categories are also characterized by these attributes, through specific surveys. It means that every category is mapped to one or more of the labels, as a place that would be selected by such a customer.

It is actually a kind of segmentation of the venue categories, and it is important to use the same features of the customer profile. The scoring will be calculated by multiplying the categories-characteristics matrix by the customer profile vector

3. Methodology

3.1. Analyzing neighborhood categories

After getting the list of areas according to the administrative division and enriching the data frame with all venues from Foursquare, we would like to find which categories are more representative for every neighborhood.

The chosen method is counting the existing venues for each category, to select only the ones with the higher values. The first benefit is removal of outliers, categories with a low number of units, or just one, but the final objective is to reduce the number of categories, leaving only those that are representative, describing the type of neighborhood.

The steps are the following:

a) **Add a hot encoding for categories** to every venue entry

   A new dataframe is created, with venues in rows and categories in columns. The

neighborhood is moved to the first column

b) **Group by neighborhood**, calculating the mean of occurrences in the categories

The result is a modified dataframe with one row for each neighborhood, and the count of venues in the column corresponding to every category

c) **Sorting columns** in every row according to the count value for every category, and **selecting just the top 10 numbers**

The result is a new dataframe, with one row for each neighborhood, and 10 columns with the categories with the highest number of venues

3.2. Preparing preferences information

Once having the neighborhood profiles with the most describing venue categories, we need a customer preferences reference, mapped to venue categories. It will allow to score every neighborhood, according to its similarity to the customer values.

Three steps are required:

a) Customer lifestyle and preferences

A fixed dataframe is created from a list of describing attributes of the customer. It has only one row, with the characteristics as columns. A boolean value is valid to select which attributes are considered for the individual, but using a 0-1 binary format is interesting. It allows to use multiplication, where columns with zero values will not contribute to the total score.

	YOUNG	MATURE	OLD	SINGLE	MARRIED	CHILDREN	PETS	MUSIC	\
PREFERENCES	0.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0	

	SPORTS	CULTURE	FASHION	GOING_OUT	HEALTH	RELIGIOUS
PREFERENCES	0.0	0.0	0.0	0.0	1.0	1.0

In the actual process, this data should be provided externally, being generated by asking directly to the customer for its own personal information and likings. It is just the specific profile that categorizes the customer way of life with a list of labels, to be mapped to the Foursquare categories.

b) Customer segmentation and preferences

Another dataframe is loaded with a segmentation information file according to the

same list of attributes.

It stablishes which categories are the preferred depending on the customer segment. For example, a young couple with small children will prefer surroundings with primary schools, parks, playgrounds, etc. but not so much a night-life place, with many pubs or concerts, likely noisy.

It has one row for every different category, with the characteristics as columns. Boolean values, also, to indicate which categories are to be included and which are not. However, it is interesting again a format of a 0-1 binary value to use multiplication. Columns with zero values are not contributing to the total score.

CATEGORY	YOUNG	MATURE	OLD	SINGLE	MARRIED	CHILDREN	PETS
Dessert Shop	1.0	1.0	1.0	0.0	1.0	1.0	0.0
Pharmacy	0.0	1.0	1.0	0.0	0.0	1.0	0.0
Ice Cream Shop	1.0	1.0	1.0	1.0	1.0	1.0	0.0
Caribbean Restaurant	1.0	1.0	0.0	1.0	0.0	0.0	0.0
Donut Shop	1.0	1.0	1.0	1.0	1.0	1.0	0.0

CATEGORY	MUSIC	SPORTS	CULTURE	FASHION	GOING_OUT	HEALTH
Dessert Shop	0.0	0.0	0.0	0.0	1.0	0.0
Pharmacy	0.0	0.0	0.0	0.0	0.0	1.0
Ice Cream Shop	0.0	0.0	0.0	0.0	1.0	0.0
Caribbean Restaurant	0.0	0.0	1.0	0.0	1.0	0.0
Donut Shop	0.0	0.0	0.0	0.0	1.0	0.0

CATEGORY	RELIGIOUS
Dessert Shop	0.0
Pharmacy	0.0
Ice Cream Shop	0.0
Caribbean Restaurant	0.0
Donut Shop	0.0

Even when it is basically a fixed matrix, periodical update is recommended. Not only because of changes on Foursquare categories, but also to follow evolution of lifestyles. Of course, new labels can be included to map other customer groups that may be interesting.

c) Mapping preferences to categories

The next step is mapping the customer preferences to categories. For it, we will use multiplication (cell by cell), of the customer preferences vector, with the

segmentation-categories matrix.

As detailed previously, a zero value in customer preferences will disable the column associated to a specific profile attribute, and categories associated to this customer segment will not be considered. As an example, a null value in the ‘pets’ attribute means that venue categories demanded by the pet-friends segment will not be considered.

The multiplying function is called using lambda functionality row by row (axis=1), with the customer preferences as scalar vector.

The result is a dataframe with 428 unique categories: The columns associated to attributes that are not selected in customer preferences are forced to zero (marked with a red star), to be discarded in scoring, while the other columns carry the result values from the segment survey responses.

	CATEGORY	YOUNG	MATURE	OLD	SINGLE	MARRIED	CHILDREN	PETS	\
0	Dessert Shop	0.0	1.0	0.0	0.0	0.0	0.0	0.0	
1	Pharmacy	0.0	1.0	0.0	0.0	0.0	0.0	0.0	
2	Ice Cream Shop	0.0	1.0	0.0	1.0	0.0	0.0	0.0	
3	Caribbean Restaurant	0.0	1.0	0.0	1.0	0.0	0.0	0.0	
4	Donut Shop	0.0	1.0	0.0	1.0	0.0	0.0	0.0	

	MUSIC	SPORTS	CULTURE	FASHION	GOING_OUT	HEALTH	RELIGIOUS
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	1.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0

3.3. Calculating neighborhood scores

Now, we have to get scores for all the listed neighborhoods, to choose the most appropriates for the customer.

It uses a specific function the get the score for a single neighborhood, function that will be called for each of them in an iterative way. The inputs for the scoring function are two previously calculated dataframes:

- The most representative categories in the area, according to the number of venues
- The preferences from the customer, mapped to favorite categories from the segmentation groups

The algorithm has the following steps:

1. The list of categories is iterated, adding a **weight according to the position** (the most common, the higher number). It means those categories with more existing venues will contribute more to the score.

   As an example, a category ‘Pharmacy’ with a count of 50 venues will have more influence than a ‘Donut shop’ one with just 15.

   To avoid the undesired impact of the total number of venues, the weight is just by position in the sorted list. As we had selected only the ten more representative categories, we have weights from 10 to 1.

2. A **multiplying factor** is calculated for each category, adding all the individual attribute flags in the preferences. It means a category with more occurrences will also contribute more to the final score.

   As an example, a specific category ‘Park’ matching the attributes ‘Children’, ‘Pets’ and ‘Health’ (according to the customer lifestyle) will have more influence than a category ‘Museum’ matching only the attribute ‘Culture’.

3. **Adding all the individual category scores** will give the total score for a single neighborhood.

   The score for a single category is just multiplying the position weight by the calculated factor. Once we have all the categories scores, the final score for the area is a total sum of all values.

	Name	Score
0	Allerton	175.0
1	Annadale	165.0
2	Arden Heights	95.0
3	Arlington	85.0
4	Arrochar	175.0

. . .

The result is a new neighborhoods dataframe, sorted by scores. We want to present the best ones according to the scoring algorithm, but offering also some opportunity for customer choice. So up to 5 neighborhoods with the higher values are selected to be presented as final outcome.

3.4. Presentation

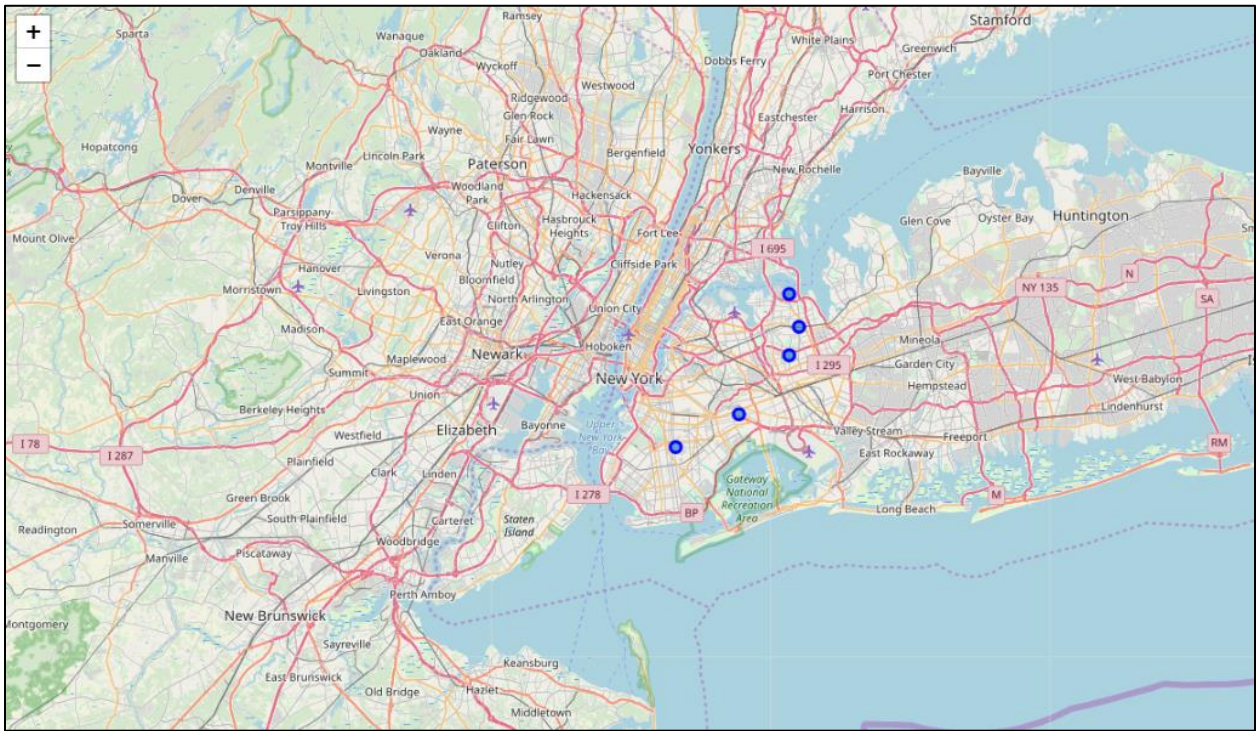
Finally, coordinates are added to the selected neighborhoods to be represented in a

map. Even when the information is the same, the usual customer will find a better experience from a graphical representation.

	Name	Score	Latitude	Longitude
0	City Line	225.0	40.678570	-73.867976
1	Pomonok	225.0	40.734936	-73.804861
2	Erasmus	225.0	40.646926	-73.948177
3	Auburndale	225.0	40.761730	-73.791762
4	Beechhurst	215.0	40.792781	-73.804365

To build the map, the Geopy library is used to get the city coordinates, and Folium to render the geographical image.

The neighborhoods are set on the map as location points, and the labels are configured to include the final position and the score.



4. Conclusions

4.1. Results and discussion

The algorithm works in a predictable way, taking a short time to get the venues for all the areas, map preferences to categories, calculate a score for every neighborhood and

build a map to show the selected ones.

There are several parameters (constants in the python script) than can be altered to get different results:

- **Radius**, the maximum distance from the location coordinates, when looking for venues.

  The default value is 500 m., a fair walking distance to understand proximity. But increasing it may be required for big areas, or zones with a low density of venues.

- **Number of venues**, to limit the list of venues retrieved from Foursquare.

  The default value is 100, which seems enough to categorize a neighborhood. In most of them, this limit is not reached.

- **Number of categories**, to limit how many will be selected as the most representative for a neighborhood.

  The default value is 10. Note that the sorting and selection is by the number of venues for each of them, and that only this short list of relevant categories will be scored.

  In most cases, a high percentage of the retrieved venues is included in this list of categories, but it means some of them that could be important for specific customers are discarded.

  It may be increased to give further detail in the description of the zone, and there is no impact on the algorithm (just more calculations).

- **Selected neighborhoods**, to reduce the results to a short list of places, the ones with the best scores

  The default value is 5, which seems to be a good number to me presented to the customer, for a personal choice.

From these parameters, and including weighting by category significance (position depending on the number of venues), the final scores are well enough differentiated to discriminate the neighborhoods. It allows to select some of them as the most appropriate for the customer satisfaction, having the highest correlation with the provided preferences.

The procedure will work for any customer and city, but two mandatory inputs must be

prepared in advance:

a) **A list with the administrative division** (they may be neighborhoods or other entities) for the target city or region.

   This list will be different for every place, so a good data source must be selected. If coordinates are not included, they should be added by a specific enrichment process.

   A relevant problem is the availability of this type of data depending on the country, or even for some remote, rural places. In these cases, a possibility is the manual compilation.

   In any case, it must be done only once for every new place (this administrative information is quite stable in time). A repository could be made to store the prepared list files, which can be reused for different customer requests.

b) **A segmentation file** depending on certain customer attributes and their association to categories.

   This information (related to people lifestyles and likings) will likely be different for every place, too, so specific surveys should be performed. It means some field work is required.

   As well, data should be as much descriptive as possible to discriminate the different tastes of people. Just 14 characteristics are used in this implementation, but much more can be added to increase information (new divisions, like worker/unemployed, or men/woman) or for smaller granularity (i.e., dividing sports in athlete, player or supporter, or by type of sports, like soccer, basketball, running, etc.).

   The only restriction is that the used characteristics must be exactly the same for the segmentation survey and the preferences answered by the customer.

   And again, the compilation must be done only once for every place, taking in consideration the characteristics or the venue categories are not changing. Otherwise, a new survey is required.

4.2. Evolution

The procedure can be directly applied, to the New York city or to any other place provided the division and segmentation input files. Actually, the expected evolution would be adding more and more cities to the list, to have a broader range of application

possibilities.

For evolution or improvement, the following options can be mentioned:

- The trivial possibility of adding new characteristics, that will allow a better description of lifestyles and preferences, and more accurate recommendations.

  The only problem is that people could be confused when facing too many possibilities, so some limit must be set.

- Using other types of geographical division, likely changing to smaller units (like streets).

  Even when it could lead to more precise location recommendations, there is another problem, related to the reduced number of venues. Most categories will have no occurrence, and this is more relevant for smaller streets, which could be rejected even if turning over the corner there are lots of valuable venues.

  This effect can be reduced by using a radius bigger than the division size, so the place will be scored also for close venues, even if they are not inside the geographical limits. But again, there should be also some balance to avoid similar scores for adjacent divisions.

- Different profiles could be prepared previously, to have an ordered list of recommended neighborhoods based on a calculated score.

  It means a collection of scores and recommendations for different customer profiles, so the user should only choose a specific profile, instead of listing his/her preferences.

  The procedure is basically the same, just the calculation is done in advance. As the number of different profiles depends only in the number of combinations of characteristics (not all of them are meaningful), the required storage can be known.