Data Science Professional Certificate - Capstone Project

Aime Lopez Aguilar
Feb 27th, 2021.

Moving Recommendations: From San Diego to London

Table of Contents:

1. Introduction
2. Data
3. Methodology
    3.1.Libraries Used:
    3.2.Data Acquisition
        3.2.1.Obtaining Neighbourhoods Lists
            3.2.1.1.San Diego
            3.2.1.2. London
        3.2.2.Obtaining Location information
        3.2.3.Visualising neighbourhoods
    3.3. Incorporating Foursquare Location data
        3.3.1. Identifying top venues by neighbourhood
        3.3.2. Data processing
    3.4. Building my preferences profile
    3.5. Applying Recommendation System
        3.5.1.Pre-processing
        3.5.2.Applying system
    3.6. Post-processing (assembling results)
        3.6.1. Collecting information for recommendations
        3.6.2. Mapping recommendations
4. Results
5. Discussion
6. Conclusion

1. Introduction:

My problem is a simple one: I'll be moving from San Diego (USA) to London (UK) soon. I would like to find out what the best neighbourhoods are to start looking for a place to live. I will leverage all the information learned in this Data Science Professional Certificate to get specific recommendations according to my tastes.

To this end I will apply a Machine Learning *Content-based Recommendation System. I will apply web scraping techniques to get information on all the communities in San Diego and London, then leverage Foursquare location data* to build community profiles. Finally I'll train the Recommendation System with my preferences, and use it to find the top ranking neighbourhoods in London that match my preference.

2. Data:

I will be using 4 main sources of data:

a) The wikipedia page for San Diego communities:
- Unlike other sources we used in this course, the html data show the community names are included as nested lists inside a table, so extracting the data will require some creativity.
- San Diego is located in Southern California, US (just north of the border with Mexico) and it has 118 Neighbourhoods

b) Nominatim from GeoPy Geolocator:
- Geopy is a Python client for several popular geocoding web services. It makes it easy for Python developers to locate the coordinates of addresses, cities, countries, and landmarks across the globe using third-party geocoders and other data sources.
- The Nominatim geocoder uses the geocoding service from OpenStreetMap
- I will be using this source to get the latitudes and longitudes for the neighbourhoods of San Diego and London

c) The wikipedia page for London neighborhoods:
- This list includes both the City of London, London, and the Greater London Metropolitan Area, totalling 531 neighbourhoods

d) Foursquare locator data:
- As we learned, Foursquare is a platform that leverages crowdsourcing to collect location information to build a dynamic database and provide useful, up-to-date answers to location-based queries such as venue information, top ranked venues around a specific location, user tips, and even trending venues.
- I will leverage Foursquare location data to identify the top venues for each Neighbourhood in San Diego and London, to build a profile for each neighbourhood based on what's available in each of them.

3. Methodology:

3.1. Libraries Used:

- pandas as pd
  - json_normalize from pandas.io.json
- numpy as np
- requests
- Nominatim from geopy.geocoders
- folium
- bs4
  - BeautifulSoup

3.2.Data Acquisition:

3.2.1. Obtaining Neighbourhoods Lists

3.2.1.1.San Diego

Source: https://en.wikipedia.org/wiki/List_of_communities_and_neighborhoods_of_San_Diego

This source presented a unique problem because the neighbourhoods are included as nested lists within a html table, rather than a simple table. Therefore, a direct scrape into a pandas dataframe was not an option. Instead, I used a loop utilising html tags for rows <tr> and elements <td> to iterate through each row of data and append data elements to a different list for each cell. I then ran each generated list through another loop to extract the content from the html format as strings and append them to a master San Diego list. This method successfully extracted all 118 communities from the website.

3.2.1.2. London

Source: https://en.wikipedia.org/wiki/List_of_areas_of_London

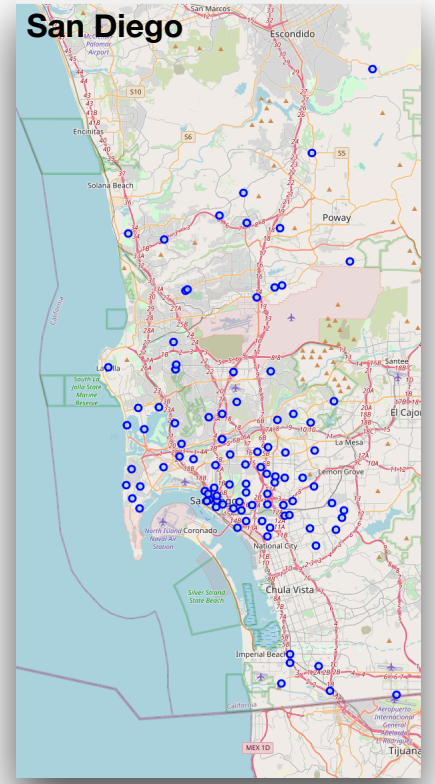
Unlike the San Diego website, the information for London neighbourhoods was stored in a table. Therefore, I was able to scrape the table and use the function read_html to convert it into a pandas dataframe directly. The resulting dataframe included columns Location (for neighbourhood name), London Borough, Post town, Dial code, and OS grid ref, for each of the 531 neighbourhoods.

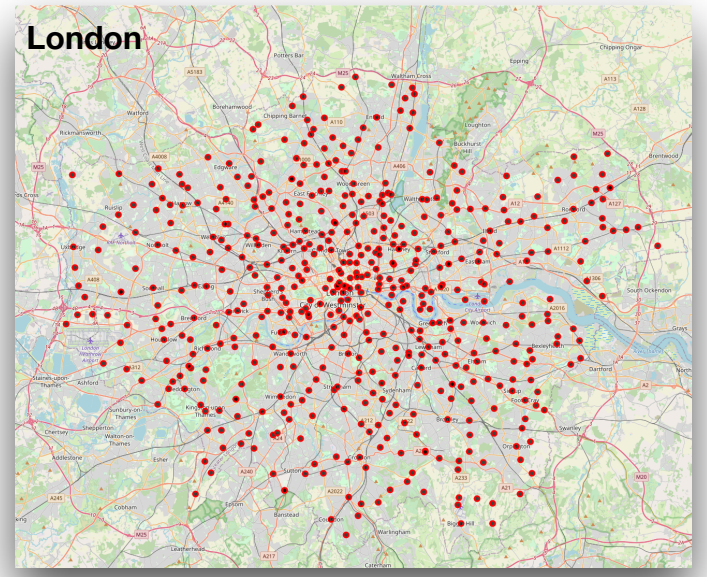
3.2.2. Obtaining Location information

Each neighbourhood name was joined with the city and country through a loop to generate a list of locations to be used as input for the Nominatim geocoder. A second loop was created to iterate through each entry of the list, running the geocoder to identify latitude and longitude for each neighbourhood and appending them to a new dataframe. The loop included a try/except clause in case the geocoder failed to assign coordinates to the neighbourhoods, in which case 'NaN' was annotated instead. After the loop was completed the neighbourhood column was added back in. Some cleanup was required after this step as Nominatim was unable to assign coordinates for some neighbourhoods. The rows without coordinates were dropped and the index was reset. The final data frames included 101 neighbourhoods for San Diego and 522 for London.

3.2.3. Visualising neighbourhoods

Maps were generated for each dataframe to visualise all the neighbourhoods in both cities. After initialising a folium map, a loop was created to iterate through each neighbourhood in the dataframe and add Circle Markers with the neighbourhood name as popups for each location.





3.3. Incorporating Foursquare Location data

3.3.1. Identifying top venues by neighbourhood

A GetNearbyVenues function was defined which looped through the dataframe, for each instance submitting explore queries to Foursquare using my client credentials and requesting the to 100 venues within 800 m of the coordinates of each neighbourhood. The results then built a new dataframe incorporating the venue information. The constructed data frames contained the following columns: Neighbourhood, Neighbourhood Latitude, Neighbourhood Longitude, Venue, Name of the venue, Venue Latitude, Venue Longitude, Venue Category.

The London loop was too large to run initially, therefore the original datagram was split into 4 smaller data frames with ~150 neighbourhoods each, and each loop was run independently, after which all resulting data frames were appended into a single dataframe with all the information.
The San Diego loop generated a dataframe with 3,463 venues identified with 324 unique categories. The London process created a dataframe with 11,314 venues identified with 413 unique categories.

3.3.2. Data processing

For later processing, each dataframe was subjected to one-hot encoding, and grouped by neighbourhood using the mean for each Venue Category to obtain the frequency of each category for every neighbourhood.

3.4. Building my preferences profile

A dataframe was created with my rankings of the top 5 and worse 5 neighbourhoods of San Diego:
A filtered database was created from the San Diego neighbourhood profile only including the neighbourhoods in my ranking. Both data frames (my rankings and the filtered San Diego profile) were sorted by neighbourhood and then were used to generate my profile. This was achieved by running a dot function between both data frames. The resulting list contained the weighted ranking for each venue category.

	Neighborhood	Ranking
0	Barrio Logan	0.1
1	East Village	0.8
2	Gaslamp Quarter	9.0
3	Hillcrest	9.9
4	Kearny Mesa	0.5
5	La Jolla	10.0
6	North Park	9.8
7	San Ysidro	0.2
8	Tijuana River Valley	0.1
9	University Heights	9.5

3.5. Applying Recommendation System

3.5.1. Pre-processing

Since some venue categories are present in San Diego but not in London, and vice versa, my preference profile and the London community profile datagrams had to be processed to include only the categories included in both. This was achieved by initiating a new clean London dataframe and running a loop through the categories in the London data. If the category was present also in my profile, the column (with all its rows of neighbourhoods) was added to the new clean London dataframe. Subsequently a new clean profile was generated in the same fashion. Using a loop through all the categories in my profile, if the category was included in the cleaned London dataframe, then it was added to the new clean profile database. This process resulted in a cleaned London dataframe and a profile dataframe with the same 273 categories.

3.5.2. Applying system

To generate the recommendations, the cleaned London dataframe was multiplied by my profile. The resulting dataframe contained the weighted categories for each neighbourhood. All categories were added and then normalised by the sum of my profile to generate a rating for all categories. Finally, the generated list was sorted according to rating to generate the top recommended neighbourhoods.

3.6. Post-processing (assembling results)

3.6.1. Collecting information for recommendations

A single dataframe was generated for the top 10 recommended neighbourhoods with all information available. First, a new function was defined to assemble the most common venue types in each neighbourhood. It first generated a new dataframe with columns for the 'nth most common venue', and then it looped through top_venues in the recommendations. For each iteration it sorted the row values in the neighbourhood profile, and added the top 10 to each column in the new row.

Finally, the latitude and longitude data from the initial location dataframe, and the newly generated dataframe with most common venues were joined to the initial recommendation table including neighbourhood name and rating. This was achieved setting the Neighbourhood column as index to generate a table with 10 neighbourhoods and 14 features (Neighbourhood, Rating, Latitude, Longitude, and 1-10th most common venues).

3.6.2. Mapping recommendations

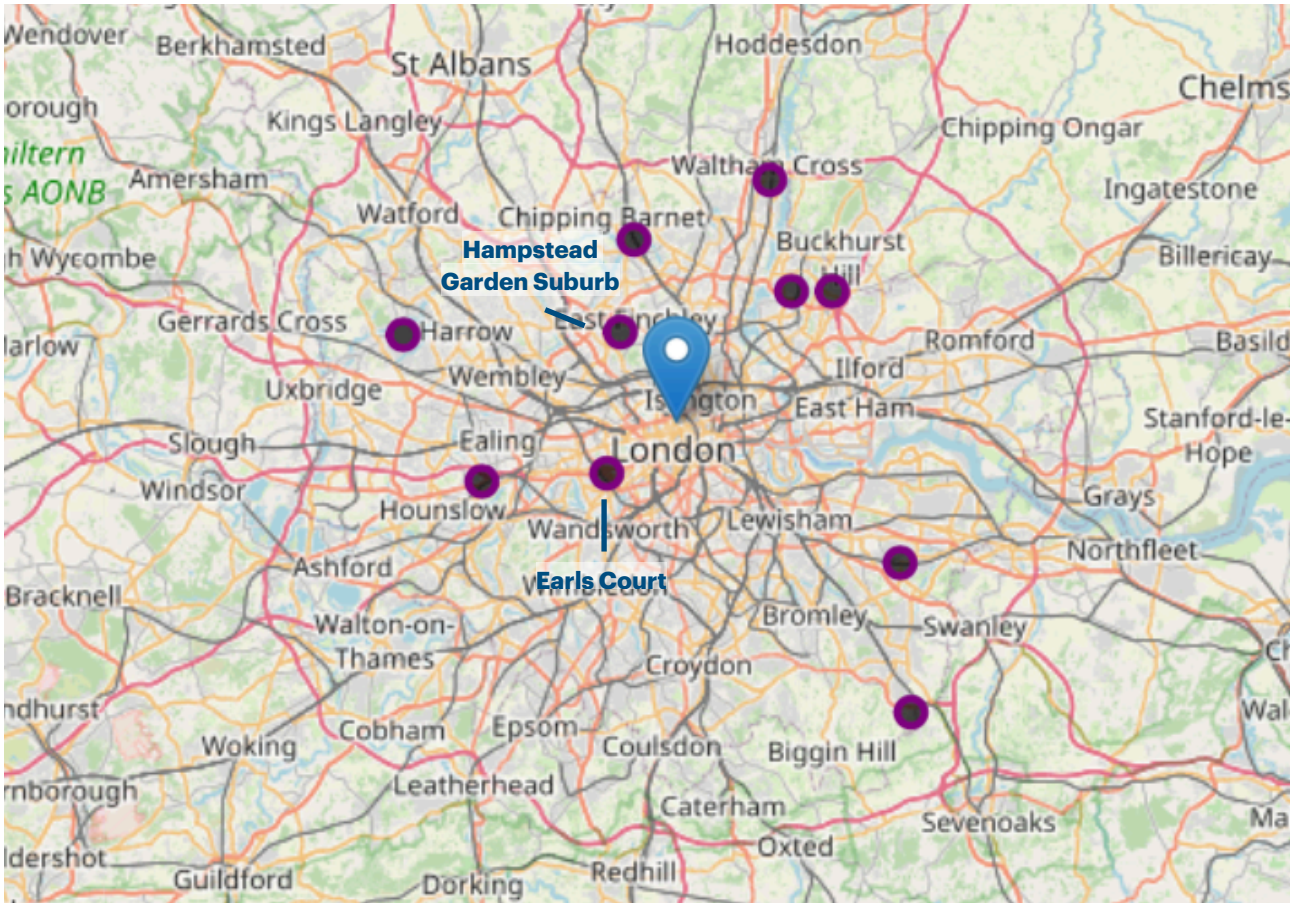
Finally a folium map was generated with only the top 10 recommended neighbourhoods. Two feature groups were added to incorporate markers. The first one was included as before (using a loop through the list of neighbourhoods to add circle markers), while the second one added a single marker at the location of my future work. The coordinates were obtained from the initial London location dataframe for the neighbourhood Bloomsbury, including a popup label for 'work'.

4. Results

Final dataframe with all information for Top 10 recommended neighbourhoods:

Neighborhood	Rating	Latitude	Longitude	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
Pratt's Bottom	0.039290	51.340884	0.111459	Bar	Coffee Shop	Zoo Exhibit	Farmers Market	Ethiopian Restaurant	Event Service	Event Space	Exhibit	Falafel Restaurant	Fast Food Restaurant
Oakleigh Park	0.028367	51.637667	-0.166225	Café	Zoo Exhibit	Farmers Market	Ethiopian Restaurant	Event Service	Event Space	Exhibit	Falafel Restaurant	Fast Food Restaurant	Cosmetics Shop
Freezywater	0.028340	51.675772	-0.031430	Café	Shoe Store	Pizza Place	Coffee Shop	Falafel Restaurant	Escape Room	Ethiopian Restaurant	Event Service	Event Space	Exhibit
Hampstead Garden Suburb	0.028139	51.580508	-0.180616	Park	Coffee Shop	Zoo Exhibit	Falafel Restaurant	Escape Room	Ethiopian Restaurant	Event Service	Event Space	Exhibit	Farmers Market
Highams Park	0.023388	51.606544	-0.008336	Gym	Coffee Shop	Zoo Exhibit	Falafel Restaurant	Escape Room	Ethiopian Restaurant	Event Service	Event Space	Exhibit	Farmers Market
Brentford	0.022952	51.486396	-0.321662	Coffee Shop	Gym	Convenience Store	Office	Furniture / Home Store	Pizza Place	Deli / Bodega	Sandwich Place	Ethiopian Restaurant	Event Service
Eastcote	0.021922	51.579542	-0.401653	Café	Hotel	Burger Joint	Coffee Shop	Sandwich Place	Grocery Store	Pharmacy	Zoo Exhibit	Exhibit	Ethiopian Restaurant
Woodford	0.021584	51.606806	0.034027	Coffee Shop	Hotel	Italian Restaurant	Café	Chinese Restaurant	Grocery Store	Restaurant	Bakery	Indian Restaurant	Pizza Place
Lamorbey	0.021457	51.435509	0.101805	Coffee Shop	Hotel	Burger Joint	Train Station	Grocery Store	Gym / Fitness Center	Dessert Shop	Restaurant	Mexican Restaurant	Pizza Place
Earls Court	0.021332	51.491612	-0.193903	Hotel	Café	Pub	Italian Restaurant	Garden	Grocery Store	Thai Restaurant	Historic Site	Pizza Place	Coffee Shop

Map of Top 10 recommended neighbourhoods with marker for future job location.



5. Discussion

*A*s can be observed, the recommended neighbourhoods spread across all London. This is a good sign, indicating that London is very diverse, i.e. it is not clustering my preferred neighbourhoods in a specific Borough. It also indicated my analysis was able to pick specific neighbourhoods that match my profile rather result in a cluster in a particular area. An advantage of this is that I can pick a neighbourhood that is close to where I'll be work in Bloomsbury(work). Based on the map, the closest recommended neighbourhoods are Hampstead Garden and Earls Court.

Analysing the information table, it seems that the criteria that contributed to higher rankings in the neighbourhood are things such as Parks, Cafes, Grocery Stores, Events venues(both Spaces, and Services), Restaurants (particularly Ethiopian :P, Falafel, and Pizza Places), Farmer Markets and Zoos... definitely all things I like, so the Recommendation system seemed to work.

Finally, from the closest recommended neighbourhoods to work, Hampstead Garden is ranked higher (#4). Furthermore, it seems to have a good spread of top categories across its top venues!

6. Conclusion

In conclusion my Machine Learning Recommendation System, trained with my preferences of the San Diego neighbourhoods, has done a good job at analysing the neighbourhood profiles in London to recommend the ones that best match my preferences.

It has helped me solve my problem by narrowing down my options and providing information on the top recommended neighbourhoods. After looking at the recommended locations close my future job, and the top venues in each of these neighbourhoods, I've selected Hampstead Garden Suburb.