Your Home Away from Home

Sameh Rasoul M.

October 27, 2020

1. Introduction

1.1. Background

The concept of leveraging Foursquare APIs to find similarities between neighborhoods and clustering them in an unsupervised matter is a fascinating one. However, a readily implementable use case for this principle may not be as straightforward for those unfamiliar with analytics. This begs the question of when, in practice, will the casual web user be interested in figuring out similarities between neighborhoods of a city?

1.2. The Problem

One problem that this may solve would be figuring out which property to select when one is relocating to an entirely different city. This is one of those times when one is faced with many questions like: Which region in the new city should s/he begin with? How to go about it? should I ask friends? or better look-up reviews? What other factors in addition to price should I be weary off? … and so on. It is a real bummer that no property listing sites has any sort of customizations depending on the user profile. Even if it does have some adaptive features; as it currently stands, no listings websites analyze which city the user is coming from and try to prioritize the listings according to their similarities to the neighborhoods in the user's home city.

1.3. The Solution

This problem hypothetically manifests itself in Toronto residents who want to take the opportunity of falling property prices in Dubai, to find themselves a new home away from home. In reality, the common 'joe' almost invariably starts at property listing websites. Property purchases are a major

investment for the majority of people, where price is a predominant deciding factor. However, none of the mainstream property listings websites have incorporated such functionality. So, to solve our problem, we will write a script that scrapes the listings results page the user is looking at, find out its locations and then cluster them together with the home city of the user. In doing so, we will have the data & findings to show the user some relevant details about the listing, similar neighborhoods back home and additional features all together in an interactive map. We will demonstrate a proof of concept by taking a resident of Toronto looking to buy an apartment in Dubai.

2. Data Acquisition & Cleaning

Several datasets will need to be scraped from multiple sources. We will divide our data acquisition and cleaning process according to the data sources, which include:

1. The property listings website results page and the pages of the individual listing in the results page
2. Wikipedia page including Toronto's postal codes and an online csv archive of geolocation of Toronto's neighborhoods.
3. Foursquare venue data for all property listings and Toronto's neighborhoods

The datasets where harvested and cleaned separately before final consolidation and exploratory analysis.

2.1. Property listings

Firstly, the dataset for the properties of interest in Dubai will be constructed as follows:

1. The user will go to Propertyfinder.ae and input his desired search parameters, then copy the URL of the results page.
2. The results page will be automatically read from the clipboard, captured as a response object; and then scraped for prices, cover pictures and hyperlink to each listing.
3. We will then iterate over the listings links getting a response object for each listing,

The data that was collected on each listing included the listing's price, the link to the listing's page, and the link to the first photo. Each results page by default had around twenty listings with occasionally one featured advert. The script to harvest this data, while also accounting for this was straightforward to write using the BeautifulSoup library. The critical missing piece of information is the geographical coordinates for each listing; and here is where the link to the listings page comes into play. By iterating over the page of each listing, were able to identify the coordinates using a suitable Regular Expression search.

In [10]:
```
# import the regular expressions library
import re

# create the regex object
coords = re.compile(r'"latitude":(\d+\.\d+),"longitude":(\d+\.\d+)')

# initialize empty lists to capture coordinates
latitudes=[]
longitudes=[]

# while iterating over the
for snip in snippets:
    mo = coords.search(snip)
    latitudes.append(mo.group(1))
    longitudes.append(mo.group(2))
```

The resulting data set requires minimal cleaning if any, and should look something like:

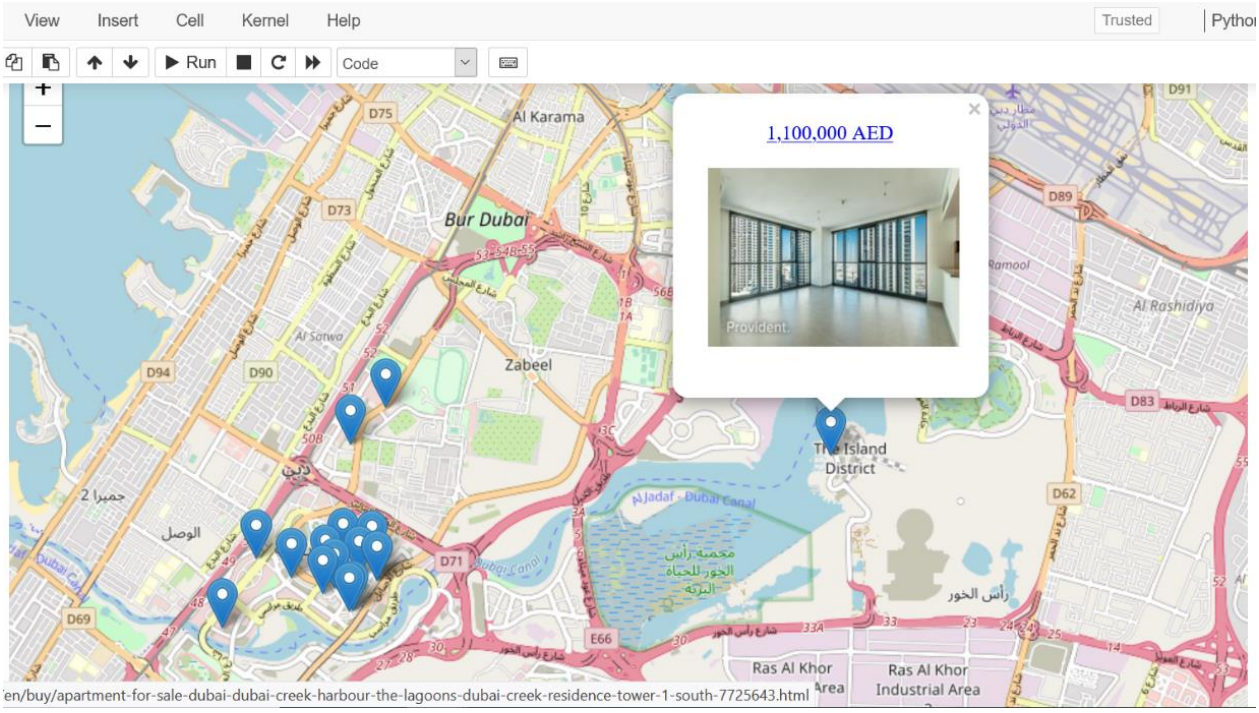
View Insert Cell Kernel Help

Run Code

```
df=pd.DataFrame({'Price':prices, 'Latitude':latitudes , 'Longitude':longitudes, 'Link':links,
df
```

	Price	Latitude	Longitude	Link	Pic
0	1,200,000 AED	25.184884	55.260565	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/dc39565...
1	1,300,000 AED	25.191404	55.279303	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/e6d7083...
2	1,190,000 AED	25.186913	55.277919	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/b097242...
3	1,000,000 AED	25.109367	55.24798	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/7c33f53...
4	1,300,000 AED	25.207533	55.277978	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/42ace08...
5	1,300,000 AED	25.189836	55.27576	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/8455cdb...
6	1,250,000 AED	25.073151	55.136982	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/f6b32e6...
7	1,049,987 AED	25.191107	55.26991	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/5990384...
8	1,000,000 AED	25.190874	55.281666	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/e16e669...
9	1,250,000 AED	25.212018	55.28286	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/c9ccbed...
10	1,110,000 AED	25.08333	55.144753	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/2ac0e08...

This dataset fulfills the requirement for mapping the properties of interest, as well as adding the functionality of displaying the cover photo of the listing and providing a hyperlink to listing in case it peaks the interest of the user. An integrity check of the gathered can be carried out by doing an initial folium map of this dataset, which includes 25 property listings:



2.2. Toronto Neighborhoods

The dataset for Toronto neighborhoods was constructed by capturing Toronto's zip codes from the Wikipedia page about the city. This provided us with the postal code of each neighborhood along with the borough name and neighborhood. The final piece required to leverage the Foursquare API is coordinates of the center of each neighborhood.

We had two options to collect this data. Option one is to use the geocoder library, which we found unresponsive and option two is to use Geopy library which was unable to pin point the location of the neighborhood and generally returned a venue somewhere in the vicinity of the neighborhood but usually not close enough to be a reliable location to identify the neighborhood center.

The situation could have been easily remedied by subscribing to the google geocoder API service; however, since this project's purpose is to provide a proof of concept, the geospatial data was

sourced from a readily available archive of the coordinates of each postal code available at http://cocl.us/Geospatial_data.

The obtained datasets from both sources were analyzed using exploratory functions like value_counts() and sample() to check for anomalies and once cleaned, the datasets were merged on the 'postal code' columns to provide all the needed neighborhoods data in one dataset that looks something like:

```
1 DF.sample(5)
```

	Postal Code	Borough	Neighbourhood	Latitude	Longitude	Link	Pic
61	NaN	NaN	1,100,000 AED	25.206125	55.343607	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/fc117ae...
54	NaN	NaN	1,199,000 AED	25.087251	55.145574	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/7ea868e...
18	M4N	Central Toronto	Lawrence Park	43.728	-79.3888	NaN	NaN
14	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.6368	-79.4282	NaN	NaN
44	NaN	NaN	1,300,000 AED	25.189836	55.27576	https://www.propertyfinder.ae//en/buy/apartmen...	https://www.propertyfinder.ae/property/8455cdb...

2.3. Foursquare

With two datasets, one including latitude and longitude values for the property listings, and the other having the same for the neighborhoods in Toronto, we can proceed with utilizing the Foursquare API to explore the venues surrounding all the locations. The datasets where concatenated and property listings prices were taken as their names.

From now on, we will be referring to either a listing in the new city or a neighbourhood in the home city as a 'location'.

3. Exploratory Analysis

Hence, taking one location at a time, the top 100 venues for each location were gathered. A sample result from the dataframe of all venues shows that the desired data is being captured, and that includes both venues from around the properties in Dubai and as well as from Toronto

neighbourhoods. We can identify the properties here as those with a price in AED in the ‘Neighbourhood column’

```
[44]: print(all_venues.shape)
      all_venues.sample(10)
```

```
(2912, 7)
```

[44]:

	Neighbourhood	Neighbourhood Latitude	Neighbourhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
2379	1,100,000 AED	25.082217	55.142624	The Radisson Blu Residence, Dubai Marina	25.078288	55.143216	Hotel
1984	1,049,987 AED	25.191107	55.26991	Gazebo	25.190844	55.266950	Indian Restaurant
641	Little Portugal, Trinity	43.6479	-79.4197	Ufficio	43.649439	-79.423014	Italian Restaurant
487	Richmond, Adelaide, King	43.6506	-79.3846	Booster Juice	43.648898	-79.383351	Juice Bar
199	St. James Town	43.6515	-79.3754	Downtown Camera	43.653107	-79.375120	Camera Store
1911	1,250,000 AED	25.073151	55.136982	Zafran Indian Bistro	25.076842	55.139459	Indian Restaurant
1419	St. James Town, Cabbagetown	43.668	-79.3677	China Gourmet	43.664180	-79.368359	Chinese Restaurant
1539	Church and Wellesley	43.6659	-79.3832	Como En Casa	43.665160	-79.384796	Mexican Restaurant
4	Regent Park, Harbourfront	43.6543	-79.3606	Impact Kitchen	43.656369	-79.356980	Restaurant
2404	1,199,000 AED	25.087251	55.145574	Habtoor Grand Resort, Autograph Collection	25.085991	55.141161	Resort

Subsequently, the frequency of occurrence of the venue categories were calculated for each location. The result, i.e. frequency of category-wise frequency of occurrence looked as follows:

Sample from Dubai Property listings of the calculated frequencies:

```
----1,299,999 AED----
                venue  freq
0                Café  0.09
1              Resort  0.06
2          Restaurant  0.06
3  Italian Restaurant  0.06
4         Coffee Shop  0.06


----1,300,000 AED----
                        venue  freq
0                 Coffee Shop  0.09
1  Middle Eastern Restaurant  0.07
2                       Hotel  0.07
3                        Café  0.05
4                  Restaurant  0.04
```

Similarly, the calculated frequencies for the neighbourhoods of Toronto look as follows:

```
----Central Bay Street----
                 venue  freq
0          Coffee Shop  0.18
1                 Café  0.06
2   Italian Restaurant  0.04
3       Sandwich Place  0.04
4  Japanese Restaurant  0.03


----Christie----
                venue  freq
0       Grocery Store  0.25
1                Café  0.19
2                Park  0.12
3         Coffee Shop  0.06
4  Italian Restaurant  0.06
```

4. Prediction Models

4.1. Classification Models

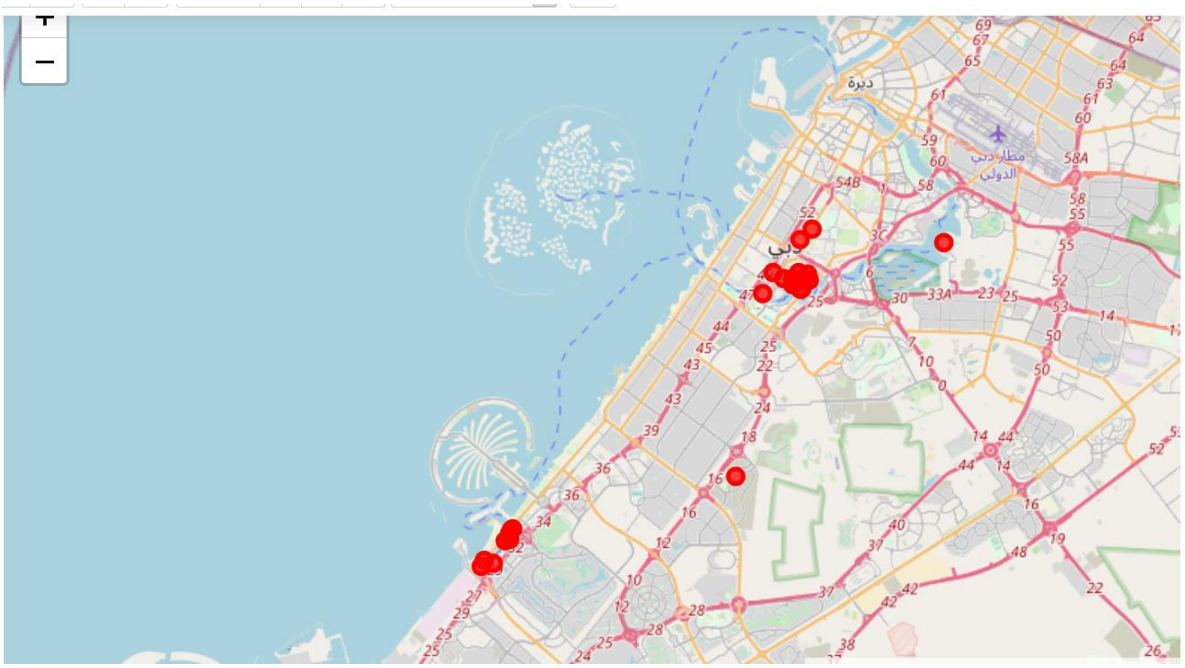
With this data, one can choose to apply one of many methods of classification if the user has the preconceived opinions in his own hometown. For example, a user might classify the neighbourhoods that he/she likes as cool, uptown, pleasant, etc. and though they don't like as trashy, noisy, gloomy, etc. Then taking on a supervised classification problem, a model could be built to predict the classification of the Dubai property listings based on the venues data.

4.2. Clustering Models

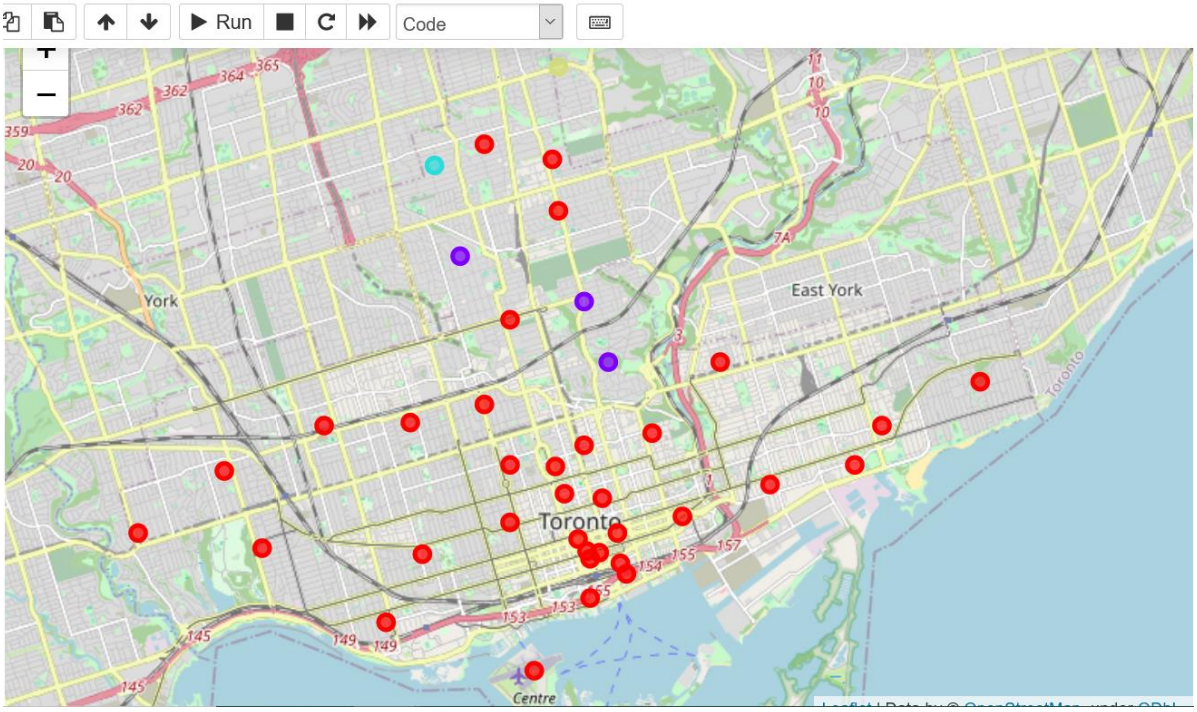
Another more straightforward approach would be to immediately apply an unsupervised clustering model on the venue data like K-means or DBSCAN. For avoiding the mistake of classifying a poor quality neighbourhood as an outlier, we avoided DBSCAN. We applied instead we a K-means clustering model to the combined dataset, which effectively clusters the Dubai property listings together with Toronto neighbourhoods.

This essentially pairs every property listing to a cluster of neighbourhoods in Toronto, meaning that if the property was to hypothetically be listed in Toronto, it will most likely belong with this group of neighbourhoods, based on its surrounding amenities and facilities, i.e. venues.

The results of clustering color codes the listings according to the cluster they were found most compatible with. Passing on our example we see that all the properties were classified together as belonging to the same cluster which is colored amber:



Zooming out and re-directing our map back to the city of Toronto we find that this cluster includes the neighbourhoods in the southern site or what might seem like ‘downtown’ Toronto:



To evaluate the performance of the model, we can also have look at the common venue types in a sample selection from our dataset, and it is evident that there is a prevalence of restaurants, coffee shops and cafes in the group.

```
locations_venues_sorted.sample(5)
```

	Cluster Labels	Neighbourhood	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
37	0	Parkdale, Roncesvalles	Breakfast Spot	Gift Shop	Cuban Restaurant	Eastern European Restaurant	Bookstore	Movie Theater	Dessert Shop	Italian Restaurant	Bar	Restaurant
29	0	Harbourfront East, Union Station, Toronto Islands	Coffee Shop	Aquarium	Hotel	Café	Restaurant	Fried Chicken Joint	Brewery	Scenic Lookout	Park	Baseball Stadium
22	0	Commerce Court, Victoria Hotel	Coffee Shop	Restaurant	Café	Hotel	Gym	American Restaurant	Japanese Restaurant	Deli / Bodega	Seafood Restaurant	Bakery
1	0	1,040,000 AED	Café	Spa	Cocktail Bar	Coffee Shop	Harbor / Marina	Chinese Restaurant	Hotel	Middle Eastern Restaurant	Sports Bar	Latin American Restaurant
10	0	1,217,819 AED	Hotel	Café	Restaurant	Middle Eastern Restaurant	Coffee Shop	Seafood Restaurant	Breakfast Spot	Lounge	Chinese Restaurant	American Restaurant

5. Conclusion

With this info, we could conclude the property listings results page passed to the program included a largely similar selection of properties similar to southern part of Toronto city. This conclusion was reached based on a predictive clustering model applied to a selection of listings on a results page from propertyfinder.ae. The model can be easily reapplied to any other results page from different cities on the same website, and easily adapted to scrape any listings website. The model can help people looking to relocate to an entire new city with identifying the similarities between new potential locations and the neighbourhoods in the home city.

6. Future Direction

The model is largely dependent on the quality of data available from the Foursquare service, and runs under the assumption that venues categories in the vicinity of the location is a good indication on the nature of living on that neighbourhoods. A good development on the project would be to expand the dataset to encompass other sources of data in addition to foursquare, such as google maps reviews, trip-advisor, etc. and also official data such as crime rate, insurance rates, etc.

It may also be the case where, other clustering models might provide a better prediction as it eliminates outliers, which may occupy a cluster by themselves. The potential possibilities and applications are numerous.