The Battle of the Toronto Neighborhoods

Finding the best spot to open a business - REPORT & PRESENTATION

[Author name] 7/29/21 Capstone Project Data Science

Contents

The Battle of the Neighborhoods – Finding the best spot for an Italian Restaurant

Introduction

The cosmopolitan Canadian city of Toronto now has more than six million and the most ethnically diverse culture in the world. But three hundred years ago, it was little more than a portage where the Humber River flows into Lake Ontario. It was known only to local natives and a few French voyageurs.

The city has consistently seen people from around the world move to the city and call it home. It has been a center for trade and economic growth. Toronto is known worldwide as a cultural melting pot. So, these various cultures combined to create great diversity for themselves. Since People from all over the world tend to come up here, we can see many of their cultural aspects Transport, Food, Clothing, ethnicity, etc.

This machine learning application, based on the Foursquare geolocation data, aims to help people migrating to exploring the Toronto neighborhoods and find the one spot that suits them the most to open an Italian Restaurant. It will help people making smart and efficient decisions on selecting neighborhoods to build a business out of a large and diverse set of neighborhoods in Toronto based on lifestyle (café, museums, clubs), on their social/family needs (friends, schools, colleges, etc.), daily shopping needs and habits (groceries, supermarkets, malls, etc.), as well as critical services (hospitals, utilities, etc.). In general, this application is developed on the philosophy that what people experience in the real world and the places they go are powerful reflections of who they are and what they care about.

In this particular scenario, the application analyzes the various features, i.e., housing prices, schooling ratings, weather conditions, crime rate, recreational activities, quality of emergency services, etc., of the Toronto neighborhoods and performs a comparative analysis between neighborhoods.

Therefore, migrants to these neighborhoods can make informed decisions before moving into a new city, state, country, or place for their work or start a new fresh life.

The problem at hand

In this scenario, the application suggests the "best" (relative to the diverse features mentioned above) Neighborhood to the person migrating or moving in there that wishes to open an Italian restaurant. It provides the person with relevant info in sorted lists of similar businesses in the region to find the best spot in a Toronto Neighborhood.

Solution Workflow

The application uses the Foursquare API as its prime data gathering source, possibly the best complete geolocation platform that employs a database with millions of places and APIs, allowing us to perform location search, location sharing, and details about a business.

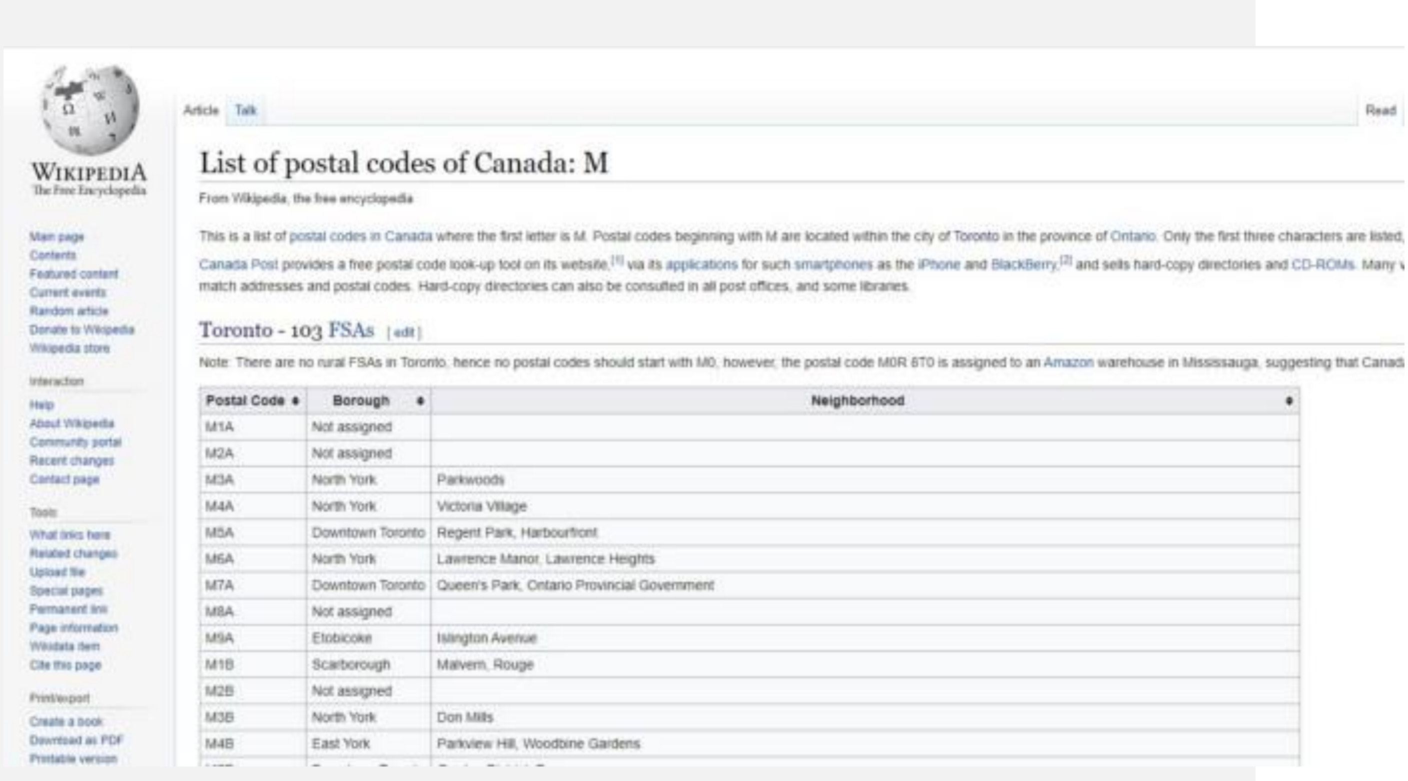
The application uses HTTP request limitations to the Four-square database, where we set a limit in the number of places per neighborhood parameter to 100 and the radius parameter to 500. The main workflow activities are:

A. Select the Data of

1. Neighborhoods
2. Neighborhoods' Latitude
3. Neighborhoods' Longitude
4. Venues
5. Name of the Venue, e.g., the name of a store or Restaurant
6. Venues' Latitude
7. Venues' Longitude
8. Venue's Category

Data acquisition

Source 1: Toronto Neighborhoods via Wikipedia



List of postal codes of Canada: M

From Wikipedia, the free encyclopedia

This is a list of postal codes in Canada where the first letter is M. Postal codes beginning with M are located within the city of Toronto in the province of Ontario. Only the first three characters are listed,
Canada Post provides a free postal code look-up tool on its website,[1] via its applications for such smartphones as the iPhone and BlackBerry,[2] and sells hard-copy directories and CD-ROMs. Many v
match addresses and postal codes. Hard-copy directories can also be consulted in all post offices, and some libraries.

Toronto - 103 FSAs [edit]

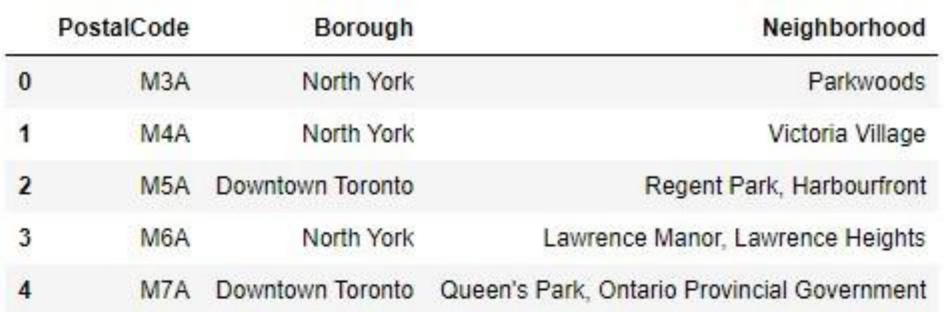
Note: There are no rural FSAs in Toronto, hence no postal codes should start with M0; however, the postal code M0R 8T0 is assigned to an Amazon warehouse in Mississauga, suggesting that Canad

Postal Code ♦	Borough ♦	Neighborhood ♦
M1A	Not assigned	
M2A	Not assigned	
M3A	North York	Parkwoods
M4A	North York	Victoria Village
M5A	Downtown Toronto	Regent Park, Harbourfront
M6A	North York	Lawrence Manor, Lawrence Heights
M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government
M8A	Not assigned	
M9A	Etobicoke	Islington Avenue
M1B	Scarborough	Malvern, Rouge
M2B	Not assigned	
M3B	North York	Don Mills
M4B	East York	Parkview Hill, Woodbine Gardens

Figure 1:Wikipedia Page showing List of Neighborhoods in Toronto with respective Postal Codes

https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M

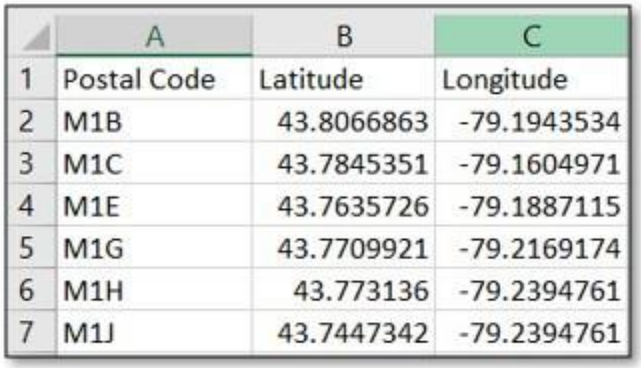
The Wikipedia site shown above provided almost all the information about the neighborhoods. It included the postal code, Borough, and the name of the neighborhoods present in Toronto. Since the data is not in a suitable format for analysis, scraping of the data was done from this site (shown in *figure2*).



	PostalCode	Borough	Neighborhood
0	M3A	North York	Parkwoods
1	M4A	North York	Victoria Village
2	M5A	Downtown Toronto	Regent Park, Harbourfront
3	M6A	North York	Lawrence Manor, Lawrence Heights
4	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

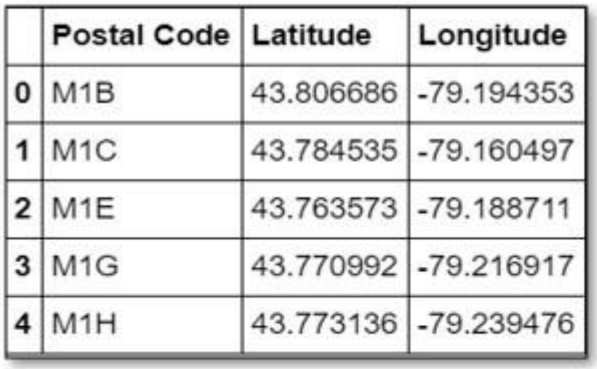
Figure 2: Data that was scraped from Wikipedia site and put into Pandas data frame

Source 2: Geographical Location data using Geocoder Package



	A	B	C
1	Postal Code	Latitude	Longitude
2	M1B	43.8066863	-79.1943534
3	M1C	43.7845351	-79.1604971
4	M1E	43.7635726	-79.1887115
5	M1G	43.7709921	-79.2169174
6	M1H	43.773136	-79.2394761
7	M1J	43.7447342	-79.2394761

Figure 4: Geographical data of Neighborhoods in Toronto



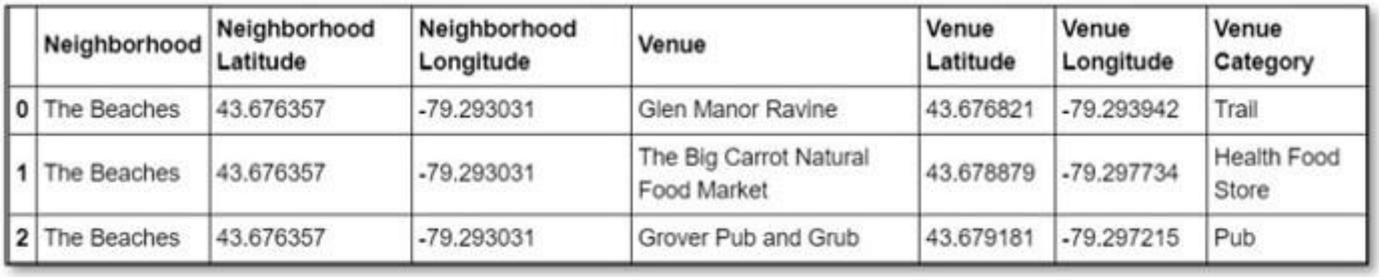
	Postal Code	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476

Figure 3: Conversion of file into Pandas data frame

2. https://cocl.us/Geospatial_data

The second source of data provided us with the Geographical coordinates of the neighborhoods with the respective Postal Codes. The file was in CSV format, so we had to attach it to a Pandas data frame(shown in figure 3).

Source 3: Venue Data using Foursquare



	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	The Beaches	43.676357	-79.293031	Glen Manor Ravine	43.676821	-79.293942	Trail
1	The Beaches	43.676357	-79.293031	The Big Carrot Natural Food Market	43.678879	-79.297734	Health Food Store
2	The Beaches	43.676357	-79.293031	Grover Pub and Grub	43.679181	-79.297215	Pub

Figure 5: Venue data pulled from Foursquare explore API

We performed a bit of data cleansing. Figure 5 (above) that the name of the neighborhood groups the neighborhoods for data clustering to be made easier later on.

4. Data Cleansing

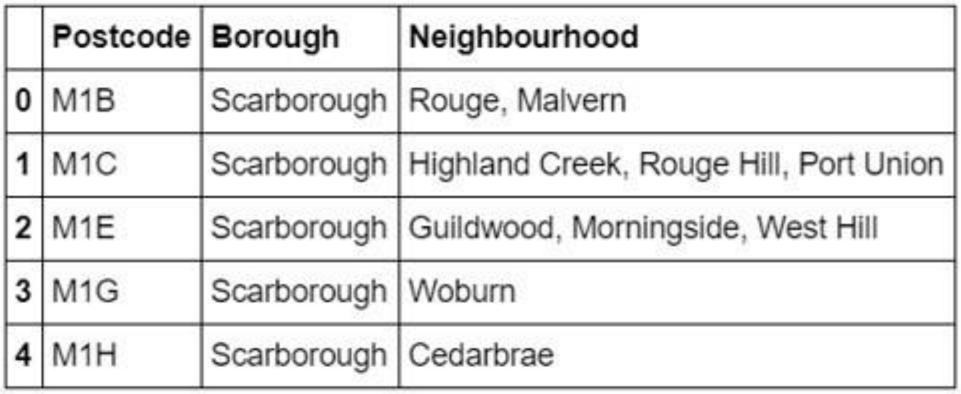
Following data collection into data frames, cleansing and merging operations were performed on the data. When getting the data from Wikipedia, some Boroughs were not assigned to any neighborhood. Therefore, the following assumptions were made:

1. Only the cells that have an assigned borough will be processed. Boroughs that were not assigned got ignored.

2. More than one Neighborhood can exist in one postal code area. For example, in the table on the Wikipedia page, you will notice that M5A is listed twice and has two neighborhoods: Harbourfront and Regent Park. These two rows will be combined into one row with the neighborhoods separated with a comma, as shown in *Figure2,* row 4.

3. If a cell has a borough but a Not assigned neighborhood, then the Neighborhood will be the same as the Borough.

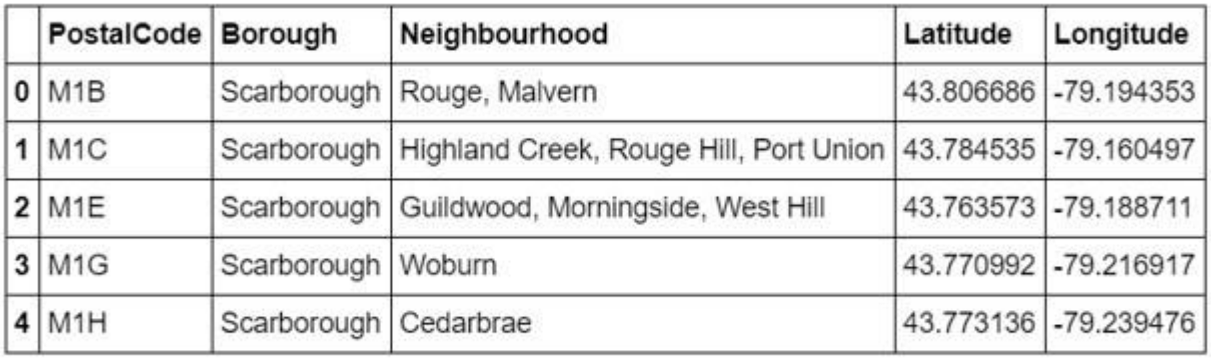
After implementing the following assumptions, the rows were grouped based on the Borough as shown below.



	Postcode	Borough	Neighbourhood
0	M1B	Scarborough	Rouge, Malvern
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union
2	M1E	Scarborough	Guildwood, Morningside, West Hill
3	M1G	Scarborough	Woburn
4	M1H	Scarborough	Cedarbrae

Figure 6: Rows grouped together based on Borough

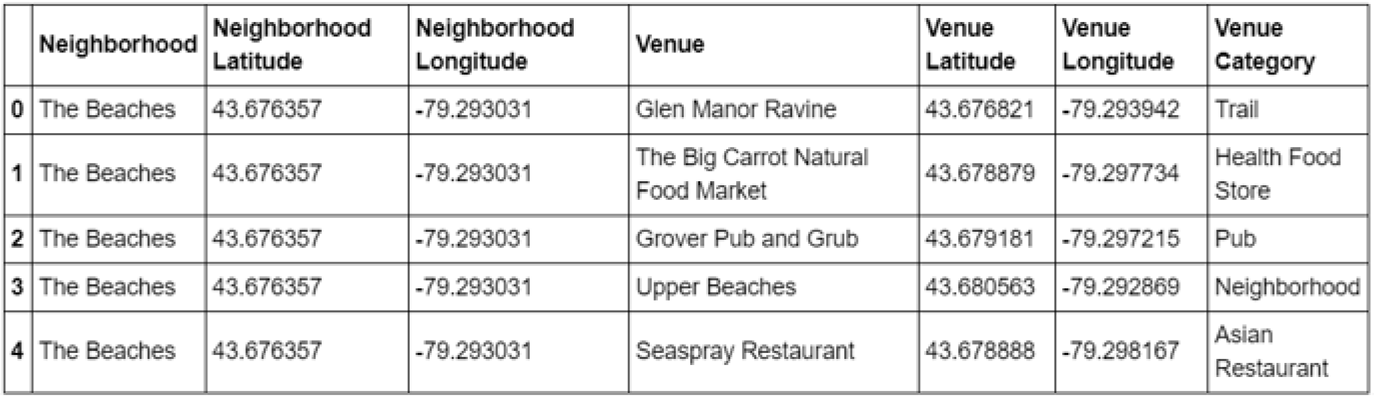
Using the Latitude and Longitude collected from the Geocoder package, we merged the two tables based on Postal Code.



	PostalCode	Borough	Neighbourhood	Latitude	Longitude
0	M1B	Scarborough	Rouge, Malvern	43.806686	-79.194353
1	M1C	Scarborough	Highland Creek, Rouge Hill, Port Union	43.784535	-79.160497
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
3	M1G	Scarborough	Woburn	43.770992	-79.216917
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476

Figure 7: Merging tables together based on Postal Code

After, the venue data pulled from the Foursquare API was merged with the table above, providing us with the local Venue within a 500-meter radius shown below.

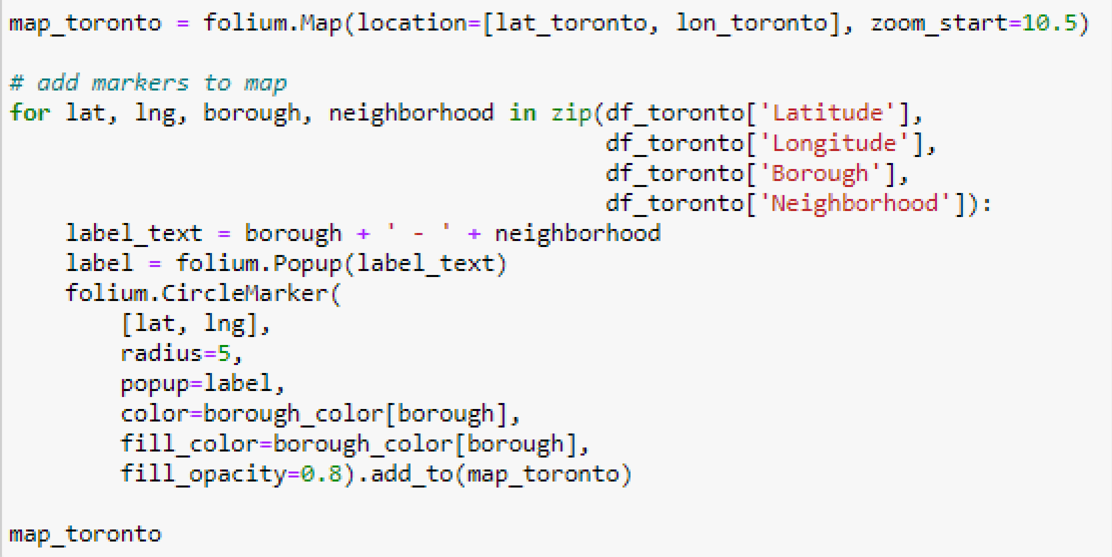


	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	The Beaches	43.676357	-79.293031	Glen Manor Ravine	43.676821	-79.293942	Trail
1	The Beaches	43.676357	-79.293031	The Big Carrot Natural Food Market	43.678879	-79.297734	Health Food Store
2	The Beaches	43.676357	-79.293031	Grover Pub and Grub	43.679181	-79.297215	Pub
3	The Beaches	43.676357	-79.293031	Upper Beaches	43.680563	-79.292869	Neighborhood
4	The Beaches	43.676357	-79.293031	Seaspray Restaurant	43.678888	-79.298167	Asian Restaurant

Figure 8: Local Venues near the respective Neighborhood

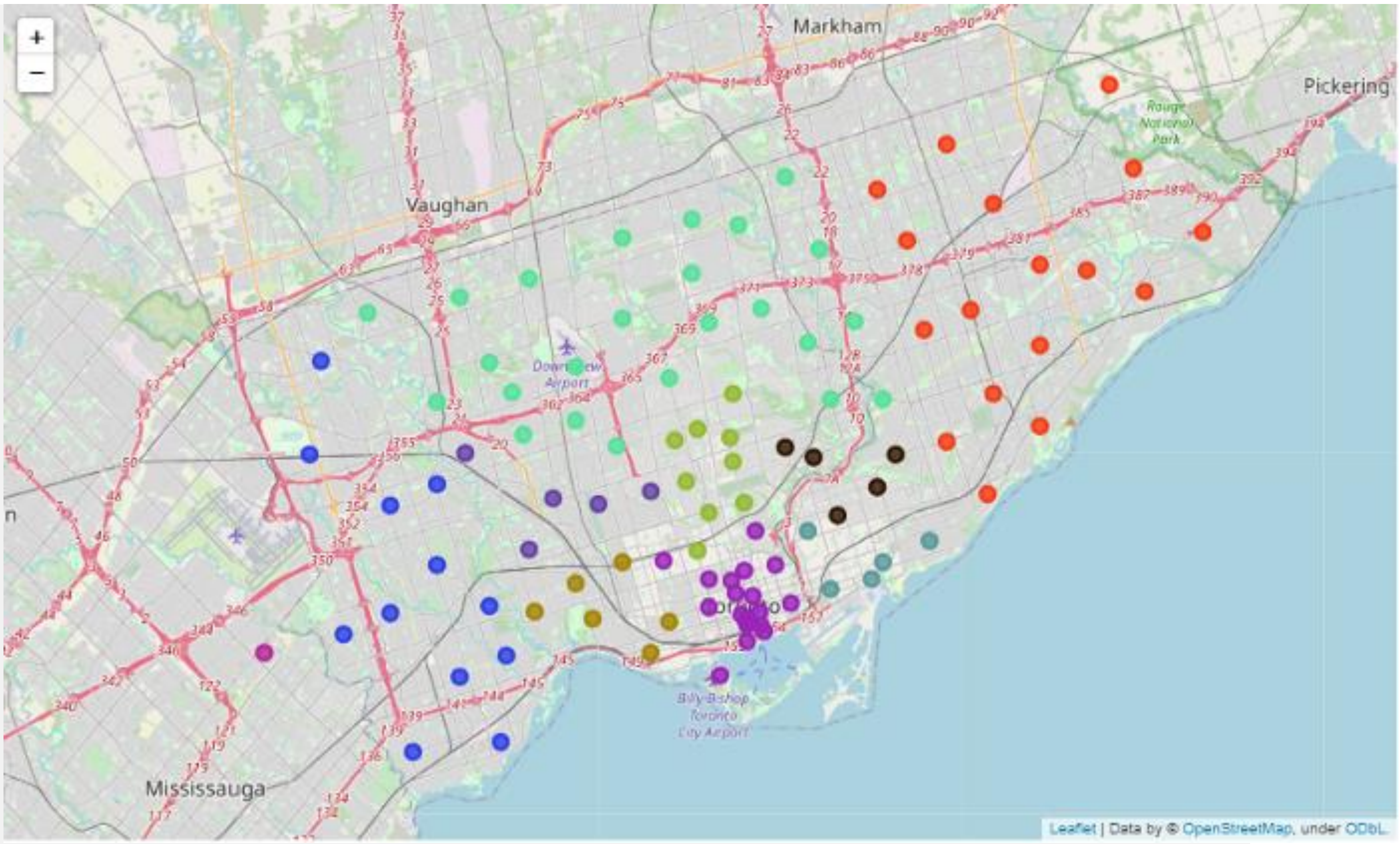
5. Data Exploration

Now, after cleansing the data, the next step was to analyze it. We then created a map using Folium and color-coded each Neighborhood depending on what Borough it was located in.

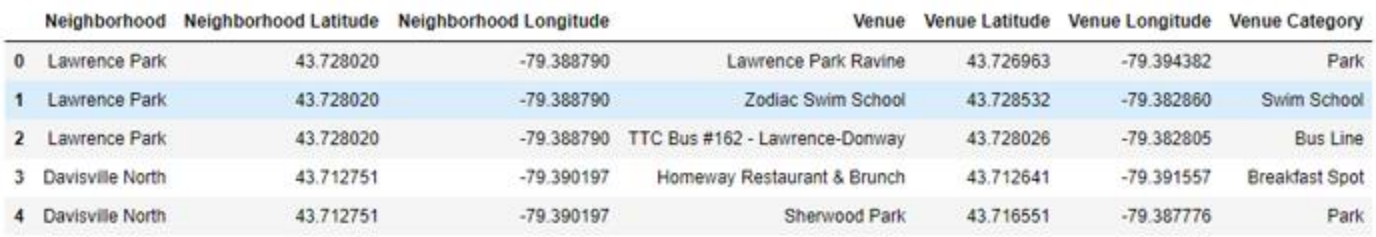


```
map_toronto = folium.Map(location=[lat_toronto, lon_toronto], zoom_start=10.5)

# add markers to map
for lat, lng, borough, neighborhood in zip(df_toronto['Latitude'],
                                           df_toronto['Longitude'],
                                           df_toronto['Borough'],
                                           df_toronto['Neighborhood']):
    label_text = borough + ' - ' + neighborhood
    label = folium.Popup(label_text)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color=borough_color[borough],
        fill_color=borough_color[borough],
        fill_opacity=0.8).add_to(map_toronto)

map_toronto
```

This snippet of code provided us with the map below:


Markham
Vaughan
Pickering
Rouge National Park
Downsview Airport
Toronto
Mississauga
Billy Bishop Toronto City Airport
+
−
Leaflet | Data by © OpenStreetMap, under ODbL.

Next, we used the Foursquare API to list all the Venues in Toronto, which included Parks, Schools, Café Shops, Asian Restaurants, etc. Getting this data was crucial to analyzing the number of Italian Restaurants all over Toronto. There was a total of 45 Italian Restaurants in Toronto. We then merged the Foursquare Venue data with the Neighborhood data, giving us the nearest Venue for each Neighborhood.

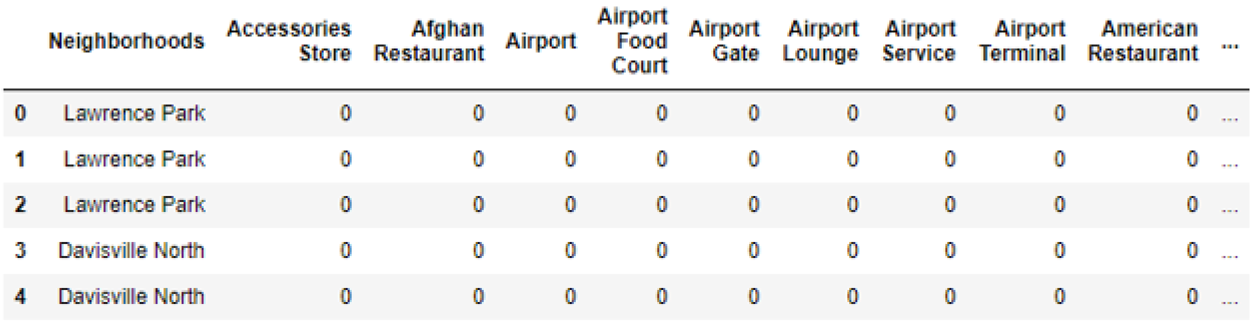


	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Lawrence Park	43.728020	-79.388790	Lawrence Park Ravine	43.726963	-79.394382	Park
1	Lawrence Park	43.728020	-79.388790	Zodiac Swim School	43.728532	-79.382860	Swim School
2	Lawrence Park	43.728020	-79.388790	TTC Bus #162 - Lawrence-Donway	43.728026	-79.382805	Bus Line
3	Davisville North	43.712751	-79.390197	Homeway Restaurant & Brunch	43.712641	-79.391557	Breakfast Spot
4	Davisville North	43.712751	-79.390197	Sherwood Park	43.716551	-79.387776	Park

Figure 10: Venue table merged with Neighborhood data

6. Data Encoding

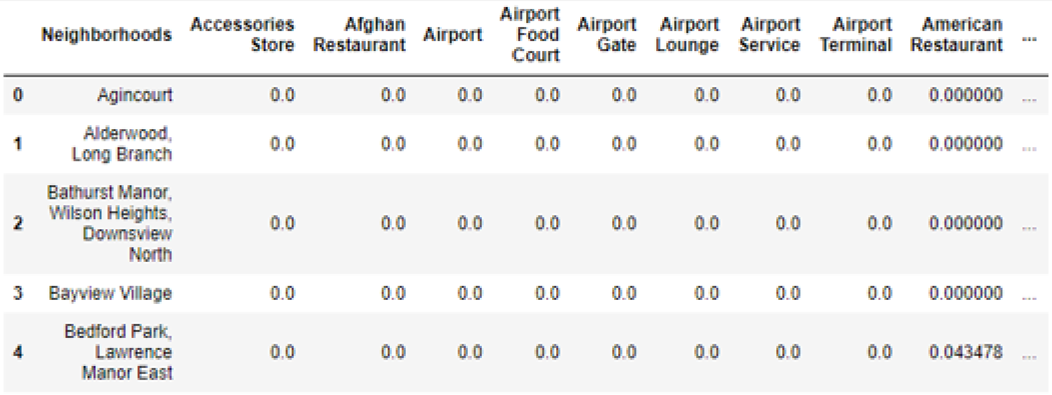
Then to analyze the data, we performed a technique in which Categorical Data is transformed into Numerical Data for Machine Learning algorithms. This technique is called **One hot encoding**. For each Neighborhood, individual venues were turned into the frequency at how many of those Venues were located in each Neighborhood.



	Neighborhoods	Accessories Store	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	...
0	Lawrence Park	0	0	0	0	0	0	0	0	0	...
1	Lawrence Park	0	0	0	0	0	0	0	0	0	...
2	Lawrence Park	0	0	0	0	0	0	0	0	0	...
3	Davisville North	0	0	0	0	0	0	0	0	0	...
4	Davisville North	0	0	0	0	0	0	0	0	0	...

Figure 11: One Hot Encoding

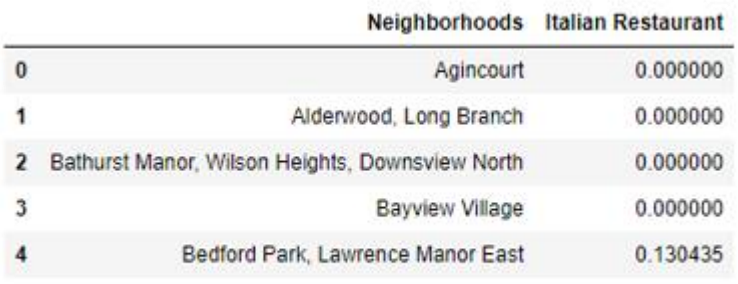
Then we grouped those rows by Neighborhood and by taking the **average** of the frequency of occurrence of each Venue Category.



	Neighborhoods	Accessories Store	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	...
0	Agincourt	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...
1	Alderwood, Long Branch	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...
2	Bathurst Manor, Wilson Heights, Downsview North	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...
3	Bayview Village	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...
4	Bedford Park, Lawrence Manor East	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.043478	...

Figure 12: Grouped Neighborhoods by the average of the frequency of each Venue

After, we created a new data frame that only stored the Neighborhood names as well as the mean frequency of Italian Restaurants in that Neighborhood. This allowed the data to be summarized based on each individual Neighborhood and made the data much simpler to analyze.



	Neighborhoods	Italian Restaurant
0	Agincourt	0.000000
1	Alderwood, Long Branch	0.000000
2	Bathurst Manor, Wilson Heights, Downsview North	0.000000
3	Bayview Village	0.000000
4	Bedford Park, Lawrence Manor East	0.130435

Figure 13: New data frame storing Neighborhoods and the average Italian Restaurant in that Neighborhood

7. Methodology

To make the analysis more interesting, we wanted to cluster the neighborhoods based on the neighborhoods that had similar averages of Italian Restaurants in that Neighborhood. To do this, we used K-Means clustering. To get our optimum K value that was neither overfitting nor underfitting the model, we used the Elbow Point Technique. In this technique, we ran a test with a different number of K values and measured the accuracy, and then chose the best K value. The best K value is chosen at the point in which the line has the sharpest turn. In our case, we had the Elbow Point at K = 4. That means we will have a total of 4 clusters.

Then we used a model that accurately pointed out the optimum K value. We imported 'KElbowVisualizer' from the Yellowbrick package. Then we fit our K-Means model above to the Elbow visualizer. That means we will have a total of 4 clusters.

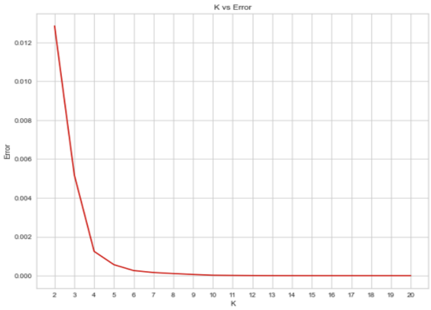


Figure 14: Finding the K vs Error Values

We just integrated a model that would fit the error and calculate the distortion score. From the dotted line, we see that the Elbow is at K=4. Moreover, in K-Means clustering, similar objects based on a certain variable are put into the same cluster. Neighborhoods that had a similar mean frequency of Italian Restaurants were divided into 4 clusters. Each of these clusters was labeled from 0 to 3 as the indexing of labels begins with 0 instead of 1.

After, we merged the venue data with the table above, creating a new table that would be the basis for analyzing new opportunities for opening a new Italian Restaurant in Toronto. Then we created a map using the Folium package in Python, and each Neighborhood was colored based on the cluster label.

Cluster 1 — Red

Cluster 2 — Purple

Cluster 3 — Turquoise

Cluster 4 — Dark Khaki

8. Cluster Analysis

We have a total of 4 clusters (0,1,2,3). Before we analyze them one by one, let's check the total amount of neighborhoods in each cluster and the average number of Italian Restaurants in that cluster. From the bar graph made using Matplotlib (figure 18), we can compare the number of Neighborhoods per Cluster. We see that Cluster 1 has the least neighborhoods (1) while cluster 2 has the most (70). Cluster 3 has 14 neighborhoods, and cluster 4 has only 8. Then we compared the average Italian Restaurants per cluster.

This information is crucial as we can see that even though there is only one Neighborhood in Cluster 1, it has the highest number of Italian Restaurants (0.1304) while Cluster 2 has the most neighborhoods but has the least average of Italian Restaurants (0.0009). The average of the average Italian Restaurant made up the data for Figure 18. Also, from the map, we can see that neighborhoods in Cluster 2 are the most sparsely populated. Now let's analyze the Clusters individually (Note: these are just snippets of the data).

Cluster 1(Red):

Cluster 1 was in the North York area. Bedford and Lawrence Manor East were the two Neighborhoods that were in that cluster. Cluster 1 had 19 unique Venue locations and out of those only 3 were Italian Restaurants. Cluster 1 had the highest average of Italian Restaurants, equating to 0.130435. The reason why the average of Italian Restaurants is the highest is that all these Restaurants are in two neighborhoods, Bedford and Lawrence Manor East.

Cluster 2 (Blue):

There was a total of 70 neighborhoods, 229 different venues, and only 1 Italian Restaurant. Therefore, the average amount of Italian restaurants near the venues in Cluster 2 is the lowest, being 0.01. The map shows that Cluster 3 was dispersed throughout Toronto, making it one of the most sparsely populated clusters.

Cluster 3 (Turquoise):

Cluster 3 had the second to lowest average of Italian Restaurants. Cluster 3 was mainly located in the Downtown area and had some neighborhoods in West Toronto, East Toronto, and North York. Neighborhoods such as Ryerson, Toronto Dominion Center, Don Mills, Garden District, Queen's Park, and many more were included in this cluster. There was a total of 176 unique venues, and out of those, 27 were Italian Restaurants.

Cluster 4 (Dark Khaki):

Cluster 4 venues were located in the Downtown, West, East and Central Toronto areas as well as Scarborough. Neighborhoods such as Central Bay Street, University of Toronto, Central Bay Street and Riverdale were some of the neighbourhoods that made up this cluster. There were a total of 91 unique Venues in Cluster 4 with 16 Italian Restaurants. This made up the second-highest average of Italian Restaurants in that cluster which was approximately 0.063.

Therefore, the ordering of the average Italian Restaurant in each cluster goes as follows:

1. Cluster 1 (≈0.1304)

2. Cluster 4 (≈0.0632)

3. Cluster 3 (≈0.0317)

4. Cluster 2 (≈0.0009)

9. Discussion:

Most of the Italian Restaurants are in cluster 1, represented by the red clusters. The Neighborhoods located in the North York area with the highest average of Italian Restaurants are Bedford Park and Lawrence Manor East. Even though there are many Neighborhoods in cluster 2, there is little to no Italian Restaurant. We see that the Downtown Toronto area (cluster 3) has the second last average of Italian Restaurants. Looking at the nearby venues, the optimum place to put a new Italian Restaurant in Downtown Toronto is as there are many Neighborhoods in the area but little to no Italian Restaurants, therefore, eliminating any competition. The second-best Neighborhoods that have a great opportunity would be Adelaide and King, Fairview, etc., which is in Cluster 2. Having 70 neighborhoods in the area with no Italian Restaurants gives a good opportunity for opening a new restaurant. Some of the drawbacks of this analysis are — the clustering is completely based on data obtained from the Foursquare API. Also, the analysis does not consider the Italian population across neighborhoods as this can play a huge factor while choosing which place to open a new Italian restaurant. This concludes the optimal findings for this project and recommends the entrepreneur to open an authentic Italian restaurant in these locations with little to no competition.

10. Conclusion

In conclusion, to end off this project, we had an opportunity on a business problem, and we solved it by utilizing numerous Python libraries to fetch the information, control the content and break down and visualize those datasets. We have utilized Foursquare API to investigate the settings in neighborhoods of Toronto, get a great measure of data from Wikipedia, which we scraped with the Beautifulsoup Web scraping Library. We also visualized utilizing different plots present in seaborn and Matplotlib libraries. Similarly, we applied AI strategy to anticipate the error given the information and utilized Folium to picture it on a map.

Ideally, this task acts as an initial direction to tackle more complex real-life problems using data science.

11. Libraries Used

Pandas: Used for creating and manipulating data frames.
Folium: used to visualize the Neighborhood's cluster distribution using an interactive leaflet map.
Scikit Learn: used for importing the k-means clustering algorithm.
JSON Library: used to handle JSON files.
XML: used to separate data from presentation, and XML stores data in plain text format.
Geocoder: used to retrieve Location Data.
Beautiful Soup and Requests: used to scrap and library to handle http requests.
Matplotlib: Python Plotting Module