Clustering Toronto neighborhood areas by surrounding restaurant type

Advay Janardhan

June 24, 2020

Introduction

Background

The Greater Toronto Area is one of the most populous in Canada, with a strong industry and many businesses. Restaurants are one aspect that contribute to the vibrant culture in the area. They are extremely important to both residents of the area and tourists visiting. Regarding tourists, restaurants are their primary food source when staying in hotels. Knowing which areas have certain types of food options for restaurants is critical for tourists.

Problem

Since restaurants have such a presence, it is extremely useful to cluster the various neighborhoods into locales with similar restaurants. This project aims to use the geographical coordinates of neighborhoods and the types of surrounding restaurants to categorize neighborhoods by type of restaurant.

Interest

For tourists, knowing which neighborhoods are similar and the characterizations of those neighborhoods is very useful. Tourists, when booking hotel reservations for the area can choose based on the surrounding restaurants, which are within walking distance. Depending on the cuisine or type of restaurant they prefer tourists can use the information to choose a hotel. Secondly, a restaurant owner opening a new shop can try to base their shop location off of this too. This tells them which neighborhoods competition may be clustered in and which neighborhoods competition is not as much. They can use the information to apply their business strategy and select an optimal opening location.

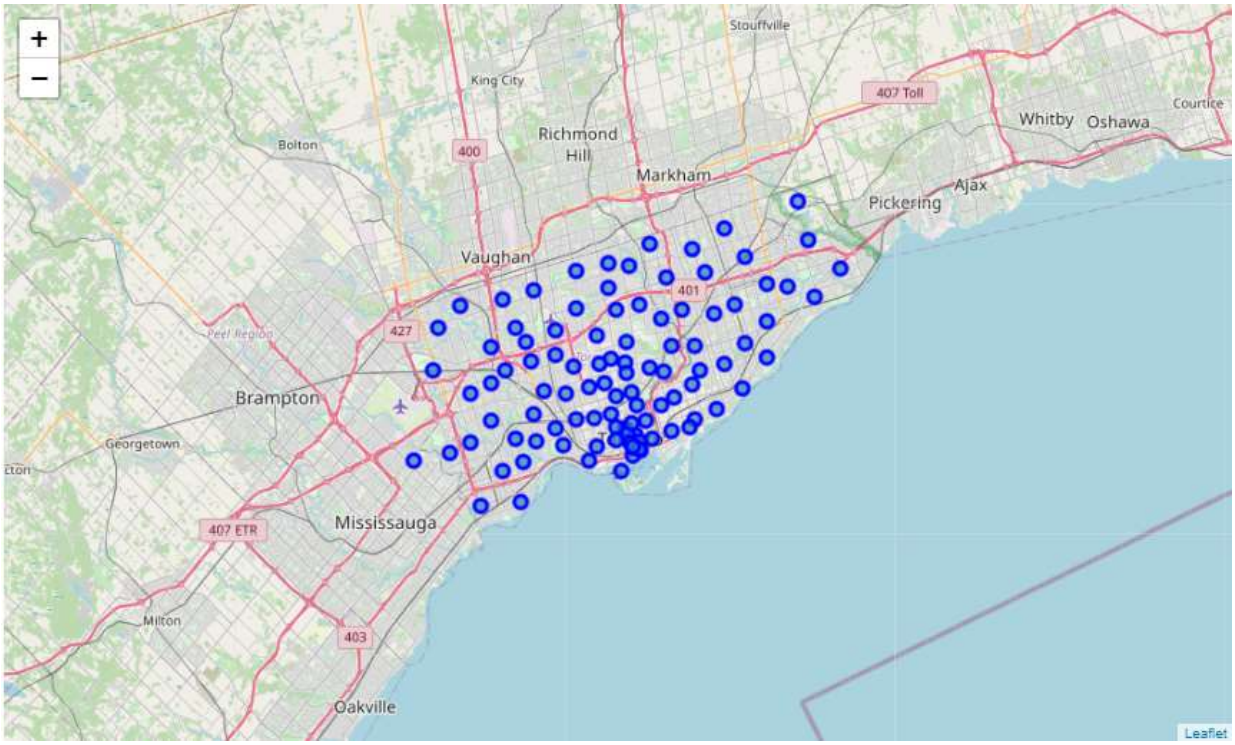
Data

The first dataset was scraped from the Wikipedia page on Toronto postal codes. This page has a list of all postal codes that start with M, giving a list of all Boroughs and Neighborhoods in Toronto. This was imported in table format with the Postal code, Borough, and Neighborhoods as columns. The second dataset came from a .csv file. This has the geographical coordinates for each postal code, which was used in conjunction with the first set. This was also imported in table format with postal code, latitude, and longitude as the columns. The third dataset was found by connecting to the FourSquare API. It yielded a list of venues within a 1000 m radius from each Neighborhood location used. This was imported by analyzing the .json file extracted from FourSquare, ultimately into a table with Neighborhoods, Neighborhood Latitude, Neighborhood Longitude, Venue, Venue Latitude, Venue Longitude, and Venue Category as columns.

Methodology

Data Processing

The first dataset of Borough and Neighborhood data had many missing Boroughs and Neighborhoods. All rows where the Borough was “not assigned” were dropped. The corresponding rows in the second dataset were also dropped and the two sets were merged into one table including postal code, Borough, Neighborhoods, Latitude, and Longitude. To gain an idea of the dataset, it was plotted as shown below:



The spacing of the area is clear and this gives an idea of how many Neighborhoods there are in addition to the dataframe. This map depicts all 103 sets of Neighborhoods.

Through connection to the FourSquare API all the coordinate values were passed into the API, which returned all the venues within 1000 m for each set of Neighborhoods. Now, because some of the locations did not return any venues, they were expunged from the analysis. This happened naturally due to empty result the API returned and did not need to be addressed. Ultimately, a dataframe with information about Neighborhoods, Neighborhood Latitude, Neighborhood Longitude, Venue, Venue Latitude, Venue Longitude, and Venue Category was constructed. To process this, the Venue Category was filtered for only restaurants. The word “restaurant” was dropped, and the column was renamed to Restaurant Type which had options such as “Chinese,” “Ethiopian,” and “Fast food.” Some entries were described broadly just as “Restaurant” and did not have any type associated with them. These were removed.

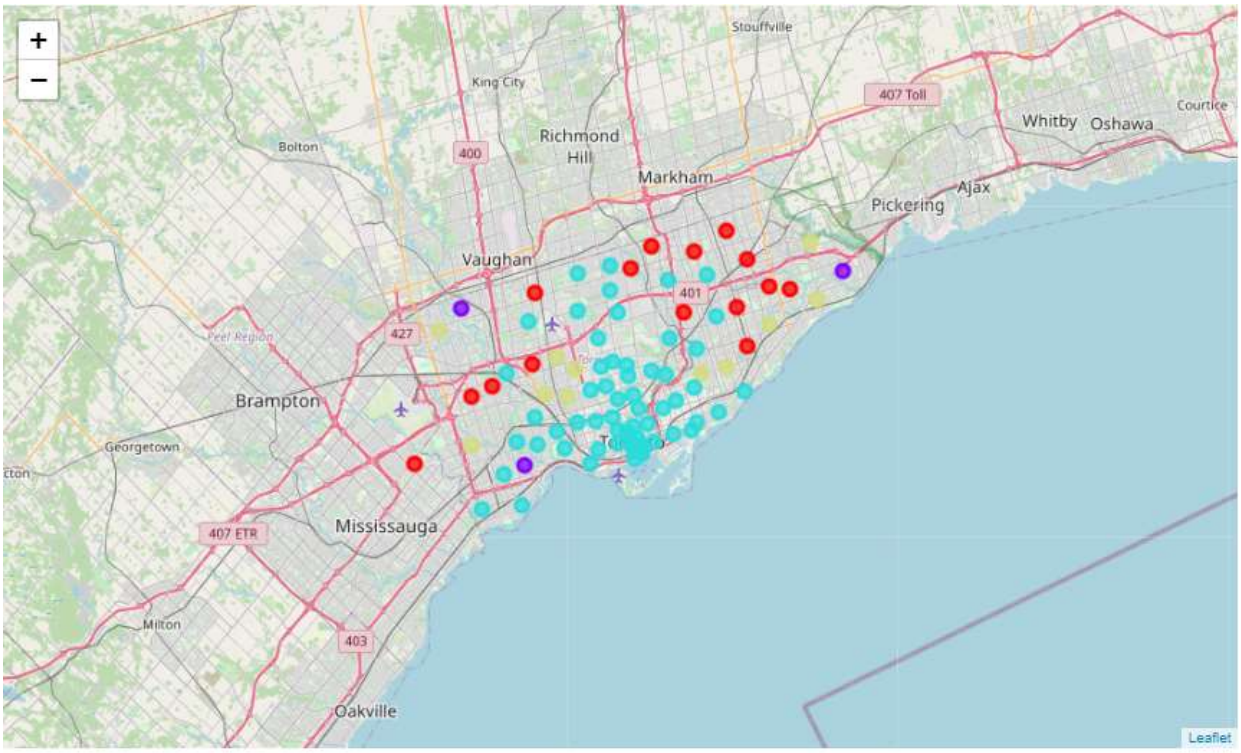
Machine Learning and Analysis

Before conducting k-means machine learning clustering on the data, one-hot encoding was performed to turn the categorical Restaurant Type column into a series of binary variables. This left the feature data set with these binary variables. The data was then grouped by Neighborhoods to enable K-means clustering. K-means clustering clusters the data by randomly picking some number of centroids, clustering by associating the points with the centroid the points are closest to, redefining the centroids as the actual centroid of the cluster, and repeating the clustering and centroid redefining process until the centroids stop moving. This will reach a local optimal solution.

K-means clustering was run using 4 clusters, and a similar map to that above was plotted. Moreover, the K-means algorithm yields cluster labels for each neighborhood. This was inserted into the binary encoded data frame that was grouped by Neighborhoods. The data frame was then filtered by each cluster (marked by 0, 1, 2, and 3), and a dictionary was created that had the Neighborhoods as keys and a list of surrounding types of restaurants as the value. For example, one entry in cluster 1 was {'Humber Summit': ['Italian']}, meaning that for Humber Summit, there was only Italian restaurants within 1000 m.

Results

The cluster map mentioned above is shown below.



From this map, it becomes clear that the Neighborhoods in downtown Toronto and Neighborhoods near those appear to be similar to each other by the blue dots. The beige colored dots appear to lie between the outskirts and the center of Toronto both in the East and in the West. They are not in the most crowded area or the least crowded area. The red dots appear to lie on the outskirts of Toronto in an arc-like shape. The purple dots appear on the very brim of the area, scattered around. Since the radius of 1000 m is a factor, it is understandable that there is some geographic correlation in the cluster, but it is very clear that coordinate geography is not the defining factor of the clusters. The final dictionary output has 4 clusters; a medium-sized cluster 0 (red on map), very small cluster 1 (purple on map), a very large cluster 2 (blue on map), and another medium-sized cluster 3 (beige on map).

Discussion

The final dictionary output that displays the Neighborhoods and associated restaurant types within the cluster is contained in the Jupyter notebook in addition to all code used. Refer to the notebook to access the output. The output shows that cluster 0 (red on map) is described by Neighborhoods that have many surrounding Asian restaurants such as Indian, Chinese, Japanese,

and Middle Eastern restaurants. Cluster 1 (purple on map) is described by Neighborhoods that mainly have surrounding Italian and Eastern European restaurants. Cluster 2 (blue on map) is described by Neighborhoods that have a little of all restaurant types. Cluster 2 neighborhoods are not characterized by specific types, but rather, the presence of a variety of types. Cluster 3 (beige on map) is described by Neighborhoods that have a lot of fast food restaurants surrounding the area.

Ultimately, a tourist that prefers Asian food should find lodging in Cluster 0 neighborhoods. A tourist that prefers Italian food should find lodging in Cluster 1. A tourist that prefers fast food should find lodging in Cluster 3. A tourist that either prefers a variety or does not know what they prefer should stay in Cluster 2 neighborhoods.

Conclusion

By using a K-means machine learning algorithm on location data extracted from FourSquare API, webpages, and a .csv file, a folium map and restaurant type dictionary were generated to characterize and group neighborhoods. This should help tourists choose lodging locations and restaurants choose building locations in Toronto. This same analysis is applicable to other large cities around the world to ideally reach similar results.