Toronto's Neighborhoods Recommender System

[Titus Chin Jun Hong]

[tituschin@tutanota.com]

+60 14 980 5623

03 December 2020

Table of Contents

1. Introduction

According to CIC News [1], Canada welcomed more than 341,000 immigrants in 2019 and Toronto has successfully attracted nearly 118,000 immigrants which contribute to almost 35% of the total number of immigrants. **The statistics indicate that most of the immigrants prefer to settle in Toronto over other cities.** Why? VisaPlace [2] has listed out 10 reasons for this question. For me, the most convincing reason is Toronto is Canada's business and financial capital, that's why immigrants prefer it.

Toronto is Canada's largest city, it has 6 boroughs which are Etobicoke, North York, East York, Central Toronto, York and Scarborough. These 6 boroughs can be further divided into 140 neighborhoods. According to City of Toronto [3], Toronto is one of the most multicultural cities in the world due to its large population of immigrants all over the world, each Toronto's neighborhood might be quite different from one another. **Therefore, out of 140 neighborhoods in Toronto, how can immigrants decide which neighborhood suits them best?** This is exactly what I want to resolve in this project.

In this project, I will try to build a Toronto's neighborhoods recommender system based on 4 factors including job opportunities, cost of living, safety and culture. So, who would be interested in this recommender system? I can say that at least 118,000 people would and I believe that this number will be growing in the future. And of course, I can't wait to find out which neighborhood suit me best too because I wish to migrate to Canada and settle in Toronto in the future. How about you?

2. Data

Previously, I mentioned that the Toronto's neighborhoods recommender system is built on job opportunities, cost of living, safety and culture. In this section, I will explain why these factors are important, describe the data that will be used and their source, import and clean the data, and finally list out all the data that will be used to create the Toronto's neighborhoods recommender system.

2.1 Factors to consider while deciding where to settle

- **Job opportunities**: We have to make a living to support ourselves or our family. And I bet we wish to get our dream job right? So, we need to know what are the common jobs for each neighborhood.
- **Affordability**: We would like to buy our dream house but how much does it cost? Curious of how much should we earn to afford to live in a specific neighborhood? To answer these questions, we need to know the affordability index for each neighborhood.
- **Safety**: We wish to live in a safe and peaceful area but how can we determine if the area is safe? To answer these questions, we need to know the crime rate for each neighborhood.
- **Culture**: We will talk and eat everyday. If possible, we would like to communicate in our favorite language and eat our favorite food right? And it's even better if our favorite things are just around us. So, it's important to know what are the language spoken most often at home and what are the popular food in each neighborhood.

2.2 Description of data and data source

Table 2.1: Description of data and data source

No.	Data	Data Description	Data Source
I.	Common jobs	These data show the common jobs for each neighborhood. The data categorize jobs according to North American Industry Classification System (NAICS) 2012. For example: professional, construction, retail trade, etc.	I extracted the data from the **2016 Toronto Neighborhood Profiles** [4]. City of Toronto uses the 2016 Canadian Census to provide a portrait of the demographic, social and economic characteristics of the people and households in each Toronto's neighborhood.
II.	Affordability index	These data show the composite Housing Affordability Index (HAI) for each neighborhood. To calculate the HAI, we need the average house price and household income data for each neighborhood.	I scraped the average house price and household income data current as of October 2020 from **Realosophy** [5]. Realosophy is a real estate brokerage company that helps their customers make better decision based on data.
III.	Crime rate	These data show the crime rate for each neighborhood.	I get the data from the **Toronto Neighborhood Crime Rates Boundary File** [6] by calling a REST API from Toronto Police Service. The file contains the 2014-2019 crime data by neighborhood.
IV.	Language spoken most often at home	These data show the language spoken most often at home in each neighborhood. For example: English, Spanish, Italian, French, etc.	I extracted the data from the **2016 Toronto Neighborhood Profiles** [4].
V.	Boundaries of neighborhoods	These data contain the boundary, latitude and longitude coordinate of each neighborhood.	I get the data from **Boundaries of Toronto's Neighborhoods** [7]. City of Toronto made the data available on its open data portal.
VI.	Popular food	These data show the popular food categories around each neighborhood according to Foursquare API. For example: Italian restaurant, Korean restaurant, Japanese restaurant, etc.	I get the data through **Foursquare API** [8]. Foursquare is a location technology platform dedicated to improve how people move through the real world.

2.3 Import data and data wrangling (optional)

In this section, I will briefly explain how to obtain the data and the steps of data wrangling for each dataframe. The codes for importing data and data wrangling can be found in my **GitHub repository** [9]. If you're not interested in data wrangling, then you can skip this part and straight away go and get all data for Toronto's neighborhoods recommender system.

I. Common jobs data

a) Import pandas library.

b) Read the 2016 Toronto Neighborhood Profiles [4] into toronto_df and clean the dataframe.

c) Extract the common jobs data for each neighborhood from toronto_df into jobs_df and clean the dataframe.

d) Define a function to convert the counts to percentage and save the data into percent_jobs_df. Figure 2.1 shows the first 5 rows of the percent_jobs_df.

	Neighborhood	ID	Accomodation	Admin/support	Agriculture	Arts	Construction	Cultural industry	Education	Finance	Health care	Management	Manufacturing	Mini
0	West Humber-Clairville	1	6.90	7.42	0.12	1.94	5.92	2.31	5.05	4.36	9.62	0.20	16.38	0
1	Mount Olive-Silverstone-Jamestown	2	8.26	7.90	0.29	1.54	6.72	2.22	3.90	3.61	9.72	0.21	19.31	0
2	Thistletown-Beaumond Heights	3	6.71	6.20	0.00	1.45	9.40	2.48	5.06	4.13	8.47	0.00	14.26	0
3	Rexdale-Kipling	4	5.49	8.90	0.47	1.70	8.14	1.80	6.34	4.17	10.13	0.00	11.55	0
4	Elms-Old Rexdale	5	6.36	7.73	0.00	1.14	6.82	2.50	5.91	4.55	11.36	0.45	11.59	0

Figure 2.1: First 5 rows of the percent_jobs_df

e) Define a function to get the top 5 common elements and save the data into top5_jobs_df. Figure 2.2 shows the first 5 rows of the top5_jobs_df.

	Neighborhood	ID	1st Most Common Job	2nd Most Common Job	3rd Most Common Job	4th Most Common Job	5th Most Common Job
0	West Humber-Clairville	1	Manufacturing	Retail trade	Transportation	Health care	Admin/support
1	Mount Olive-Silverstone-Jamestown	2	Manufacturing	Retail trade	Health care	Transportation	Accomodation
2	Thistletown-Beaumond Heights	3	Manufacturing	Retail trade	Transportation	Construction	Health care
3	Rexdale-Kipling	4	Retail trade	Manufacturing	Health care	Admin/support	Transportation
4	Elms-Old Rexdale	5	Retail trade	Manufacturing	Health care	Transportation	Admin/support

Figure 2.2: First 5 rows of the top5_jobs_df

f) Define a function to normalize the data and save the data into norm_jobs_df. Figure 2.3 shows the first 5 rows of the norm_jobs_df.

	Neighborhood	ID	Accomodation	Admin/support	Agriculture	Arts	Construction	Cultural industry	Education	Finance	Health care	Management	Manufact
0	West Humber-Clairville	1	38.673139	57.366071	33.333333	22.789116	49.160671	12.519562	32.527881	7.803618	49.187592	14.0	100.0
1	Mount Olive-Silverstone-Jamestown	2	37.378641	49.330357	66.666667	14.625850	45.083933	9.702660	20.260223	5.219638	40.177253	12.0	95.2
2	Thistletown-Beaumond Heights	3	10.517799	13.392857	0.000000	4.761905	21.822542	3.755869	9.107807	2.067183	12.112260	0.0	24.3
3	Rexdale-Kipling	4	9.385113	20.982143	41.666667	6.122449	20.623501	2.973396	12.453532	2.273902	15.805022	0.0	21.5
4	Elms-Old Rexdale	5	9.061489	15.178571	0.000000	3.401361	14.388489	3.442879	9.665428	2.067183	14.771049	8.0	17.9

Figure 2.3: First 5 rows of the norm_jobs_df

II. Affordability index data

a) Import selenium, time and BeautifulSoup libraries.

b) Define a function to scrape each neighborhood's name and website by borough from Realosophy [5] and save the data into website_df.

c) Define a function to get the average house price and household income in numerical results.

d) Scrape average house price and household income for each neighborhood into website_df.

e) Group website_df by neighborhood ID and save the data into houseprice_df.

f) Calculate the composite Housing Affordability Index (HAI) for each neighborhood by using Equation 2.1 [10].

$$Housing\ Affordability\ Index\ (HAI) = \frac{median\ household\ income}{qualifying\ income}\ x\ 100 \qquad \text{Equation 2.1}$$

g) The assumptions made to calculate the Housing Affordability Index (HAI) are:

- The average house price and household income are equal to the median house price and household income.
- The effective mortgage rates is 2% [11].
- Home buyers make a 20% down payment [12].
- The maximum monthly mortgage payment is 25% of gross monthly income.

h) If a neighborhood's Housing Affordability Index (HAI) is higher than 100, it means that most of the residents are able to afford a house in the neighborhood, the greater the HAI, the higher the housing affordability. However, if a neighborhood's HAI is lower than 100, it means that most of the residents are unable to afford a house in the neighborhood, the lower the HAI, the lower the housing affordability.

i) Save the results of Housing Affordability Index (HAI) into affordability_df.

j) Normalize the data and save the data into norm_affordability_df. Figure 2.4 shows the first 5 rows of the norm_affordability_df.

	Neighborhood	ID	HAI
0	West Humber-Clairville	1	71.577131
1	Mount Olive-Silverstone-Jamestown	2	61.091921
2	Thistletown-Beaumond Heights	3	46.788169
3	Rexdale-Kipling	4	54.670493
4	Elms-Old Rexdale	5	61.086766

Figure 2.4: First 5 rows of the norm_affordability_df

k) Convert the housing affordability index into integer. Figure 2.5 shows the first 5 rows of the affordability_df.

	Neighborhood	ID	HAI
0	West Humber-Clairville	1	112
1	Mount Olive-Silverstone-Jamestown	2	96
2	Thistletown-Beaumond Heights	3	73
3	Rexdale-Kipling	4	86
4	Elms-Old Rexdale	5	96

Figure 2.5: First 5 rows of the affordability_df

III. Crime rate data

a) Import requests and json_normalize libraries.

b) Get the crimes data from the Toronto Neighborhood Crime Rates Boundary File [6] by calling a REST API from Toronto Police Service, save the data into crime_df and clean the dataframe.

c) Calculate the crime rate for each neighborhood by using Equation 2.2 [13].

$$crime\ rate = \frac{number\ of\ crimes}{population}\ x\ 100,000\ people \qquad \text{Equation 2.2}$$

d) Normalize the data and save the data into norm_crime_df. Figure 2.6 shows the first 5 rows of the norm_crime_df.

	Neighborhood	ID	Crime Rate
0	West Humber-Clairville	1	17.713328
1	Mount Olive-Silverstone-Jamestown	2	38.429397
2	Thistletown-Beaumond Heights	3	45.357288
3	Rexdale-Kipling	4	39.298194
4	Elms-Old Rexdale	5	48.292420

Figure 2.6: First 5 rows of the norm_crime_df

e) Convert the crime rate into integer. Figure 2.7 shows the first 5 rows of the crime_df.

	Neighborhood	ID	Crime Rate
0	West Humber-Clairville	1	2857
1	Mount Olive-Silverstone-Jamestown	2	1316
2	Thistletown-Beaumond Heights	3	1115
3	Rexdale-Kipling	4	1287
4	Elms-Old Rexdale	5	1048

Figure 2.7: First 5 rows of the crime_df

IV. Language spoken most often at home data

a) Extract the language spoken most often at home data for each neighborhood from toronto_df into language_df and clean the dataframe.

b) Convert the counts to percentage and save the data into percent_language_df. Figure 2.8 shows the first 5 rows of the percent_language_df.

	Neighborhood	ID	Afrikaans	Akan (Twi)	Albanian	Amharic	Arabic	Armenian	Assyrian	Azerbaijani	Bamanankan	Belarusan	Bengali	Bikol	Bilen	Bosnian
0	West Humber-Clairville	1	0.00	0.55	0.21	0.07	0.37	0.02	0.37	0.0	0.0	0.0	0.37	0.0	0.0	0.00
1	Mount Olive-Silverstone-Jamestown	2	0.02	1.37	0.05	0.16	3.65	0.11	6.31	0.0	0.0	0.0	0.61	0.0	0.0	0.00
2	Thistletown-Beaumond Heights	3	0.00	0.74	0.06	0.06	1.25	0.06	2.62	0.0	0.0	0.0	0.57	0.0	0.0	0.00
3	Rexdale-Kipling	4	0.00	0.32	0.21	0.11	0.58	0.00	0.74	0.0	0.0	0.0	0.58	0.0	0.0	0.05
4	Elms-Old Rexdale	5	0.00	0.59	0.06	0.06	0.30	0.00	0.47	0.0	0.0	0.0	0.59	0.0	0.0	0.00

Figure 2.8: First 5 rows of the percent_language_df

c) Get the top 5 common languages for each neighborhood and save the data into top5_language_df. Figure 2.9 shows the first 5 rows of the top5_language_df.

	Neighborhood	ID	1st Most Common Language	2nd Most Common Language	3rd Most Common Language	4th Most Common Language	5th Most Common Language
0	West Humber-Clairville	1	English	Punjabi	Gujarati	Spanish	Hindi
1	Mount Olive-Silverstone-Jamestown	2	English	Punjabi	Gujarati	Assyrian	Arabic
2	Thistletown-Beaumond Heights	3	English	Punjabi	Spanish	Gujarati	Italian
3	Rexdale-Kipling	4	English	Spanish	Italian	Urdu	Croatian
4	Elms-Old Rexdale	5	English	Spanish	Somali	Italian	Vietnamese

Figure 2.9: First 5 rows of the top5_language_df

d) Normalize the data and save the data into norm_language_df. Figure 2.10 shows the first 5 rows of the norm_language_df.

	Neighborhood	ID	Afrikaans	Akan (Twi)	Albanian	Amharic	Arabic	Armenian	Assyrian	Azerbaijani	Bamanankan	Belarusan	Bengali	Bikol
0	West Humber-Clairville	1	0.0	40.789474	25.000000	11.111111	10.344828	0.363636	5.982906	0.0	0.0	0.0	4.952830	0.0
1	Mount Olive-Silverstone-Jamestown	2	50.0	100.000000	6.250000	25.000000	100.000000	2.181818	100.000000	0.0	0.0	0.0	8.018868	0.0
2	Thistletown-Beaumond Heights	3	0.0	17.105263	2.083333	2.777778	10.837438	0.363636	13.105413	0.0	0.0	0.0	2.358491	0.0
3	Rexdale-Kipling	4	0.0	7.894737	8.333333	5.555556	5.418719	0.000000	3.988604	0.0	0.0	0.0	2.594340	0.0
4	Elms-Old Rexdale	5	0.0	13.157895	2.083333	2.777778	2.463054	0.000000	2.279202	0.0	0.0	0.0	2.358491	0.0

Figure 2.10: First 5 rows of the norm_language_df

V. Boundaries of neighborhoods data

a) Import the boundary data from Boundaries of Toronto's Neighborhoods [7] into boundary_df and clean the dataframe.

b) Import geocoder and plotly libraries.

c) Visualize the boundary of each neighborhood on toronto_map. Figure 2.11 shows the toronto_map.

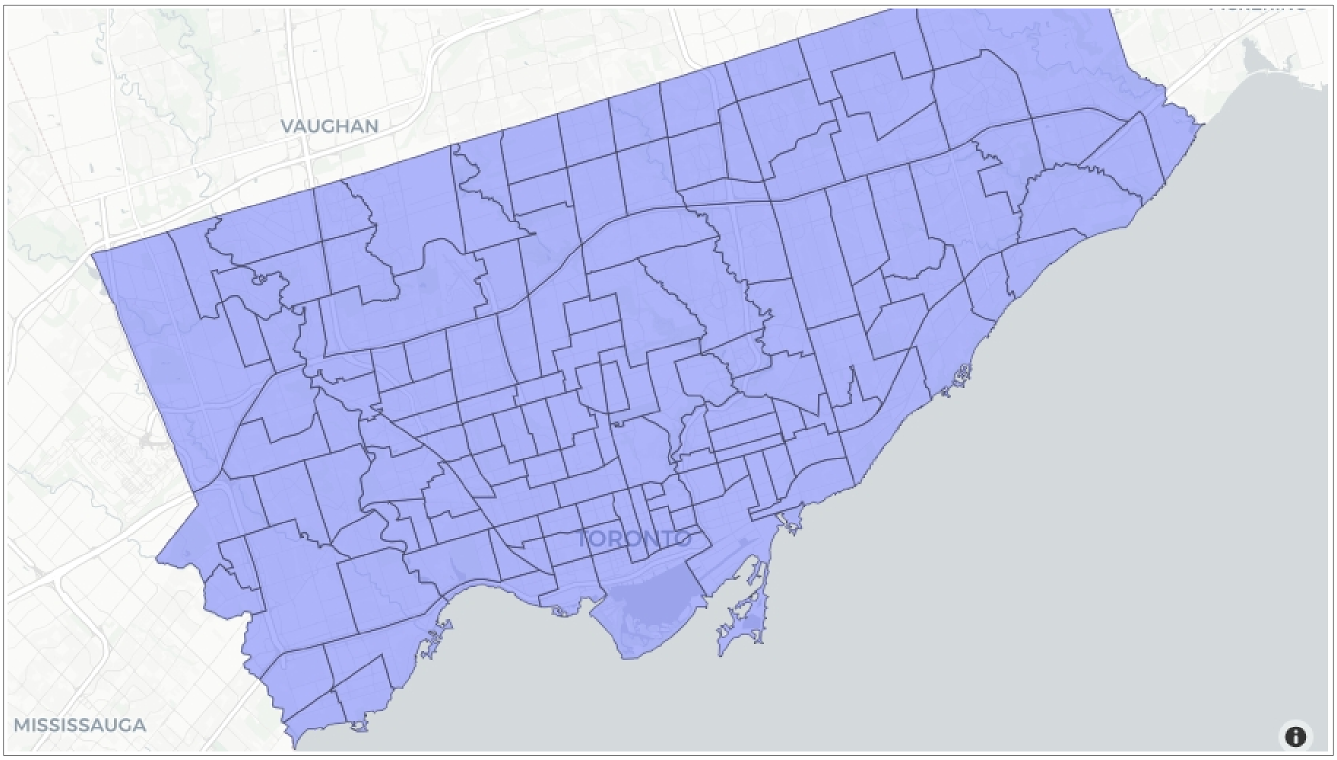


Figure 2.11: The toronto_map

VI. Popular food data

a) Get the popular food categories for each neighborhood using Foursquare API [8] into food_df and clean the dataframe.

b) Convert the counts to percentage and save the data into percent_food_df. Figure 2.12 shows the first 5 rows of the percent_food_df.

	Neighborhood	ID	Afghan Restaurant	African Restaurant	American Restaurant	Asian Restaurant	BBQ Joint	Bagel Shop	Bakery	Bistro	Brazilian Restaurant	Breakfast Spot	Burger Joint	Burrito Place	Café	Cajun / Creole Restaurant
0	West Humber-Clairville	1	2.0	0.0	4.0	4.0	0.0	0.0	2.0	0.0	0.0	2.0	6.0	0.0	0.0	0.0
1	Mount Olive-Silverstone-Jamestown	2	0.0	0.0	0.0	6.0	2.0	0.0	4.0	0.0	0.0	0.0	4.0	0.0	2.0	0.0
2	Thistletown-Beaumond Heights	3	0.0	0.0	0.0	6.0	0.0	0.0	2.0	0.0	0.0	2.0	4.0	0.0	2.0	0.0
3	Rexdale-Kipling	4	0.0	0.0	2.0	6.0	0.0	0.0	2.0	0.0	0.0	2.0	4.0	0.0	2.0	0.0
4	Elms-Old Rexdale	5	0.0	0.0	2.0	6.0	0.0	0.0	6.0	0.0	0.0	2.0	4.0	0.0	2.0	0.0

Figure 2.12: The first 5 rows of the percent_food_df

c) Get the top 5 common food categories for each neighborhood and save the data into top5_food_df. Figure 2.13 shows the first 5 rows of the top5_food_df.

	Neighborhood	ID	1st Most Common Food	2nd Most Common Food	3rd Most Common Food	4th Most Common Food	5th Most Common Food
0	West Humber-Clairville	1	Indian Restaurant	Sandwich Place	Restaurant	Chinese Restaurant	Fried Chicken Joint
1	Mount Olive-Silverstone-Jamestown	2	Sandwich Place	Indian Restaurant	Pizza Place	Italian Restaurant	Asian Restaurant
2	Thistletown-Beaumond Heights	3	Sandwich Place	Pizza Place	Indian Restaurant	Vietnamese Restaurant	Fried Chicken Joint
3	Rexdale-Kipling	4	Indian Restaurant	Chinese Restaurant	Restaurant	Sandwich Place	Steakhouse
4	Elms-Old Rexdale	5	Sandwich Place	Chinese Restaurant	Vietnamese Restaurant	Asian Restaurant	Indian Restaurant

Figure 2.13: The first 5 rows of the top5_food_df

d) Normalize the data and save the data into norm_food_df. Figure 2.10 shows the first 5 rows of the norm_food_df.

	Neighborhood	ID	Afghan Restaurant	African Restaurant	American Restaurant	Asian Restaurant	BBQ Joint	Bagel Shop	Bakery	Bistro	Brazilian Restaurant	Breakfast Spot	Burger Joint	Burrito Place	Café	Res
0	West Humber-Clairville	1	100.0	0.0	66.666667	66.666667	0.0	0.0	12.5	0.0	0.0	20.0	50.000000	0.0	0.000000	
1	Mount Olive-Silverstone-Jamestown	2	0.0	0.0	0.000000	100.000000	25.0	0.0	25.0	0.0	0.0	0.0	33.333333	0.0	7.142857	
2	Thistletown-Beaumond Heights	3	0.0	0.0	0.000000	100.000000	0.0	0.0	12.5	0.0	0.0	20.0	33.333333	0.0	7.142857	
3	Rexdale-Kipling	4	0.0	0.0	33.333333	100.000000	0.0	0.0	12.5	0.0	0.0	20.0	33.333333	0.0	7.142857	
4	Elms-Old Rexdale	5	0.0	0.0	33.333333	100.000000	0.0	0.0	37.5	0.0	0.0	20.0	33.333333	0.0	7.142857	

Figure 2.14: The first 5 rows of the norm_food_df

2.4 All data for Toronto's neighborhoods recommender system

All data needed to build a Toronto's neighborhoods recommender system are listed below. For easier replication of this project, I uploaded all data to my **GitHub repository** [9].

- percent_jobs_df
- top5_jobs_df
- norm_jobs_df
- affordability_df
- norm_affordability_df
- crime_df
- norm_crime_df
- percent_language_df
- top5_language_df
- norm_language_df
- toronto_geojson
- percent_food_df
- top5_food_df
- norm_food_df

With these data, we are ready to build a Toronto's neighborhoods recommender system! Let's get started!

3. Methodology

I wish the recommender system can be a web-based interactive dashboard, so that more people can access to it. To achieve this, I will first build a recommender system to automatically rank the neighborhoods based on our preferences. After that, I will use some attractive plots to visualize the results. Then I will convert the recommender system to an interactive dashboard. If everything is cool, then I will deploy my interactive dashboard on Heroku and guess what? A Toronto's neighborhoods recommender system is born! I have uploaded the full codes to my **GitHub repository** [9], therefore I will just briefly explain the steps for building this recommender system.

3.1 Build a recommender system

To build a recommender system, we will need to define some functions to automatically rank the neighborhoods based on our preferences.

a) Import pandas library.

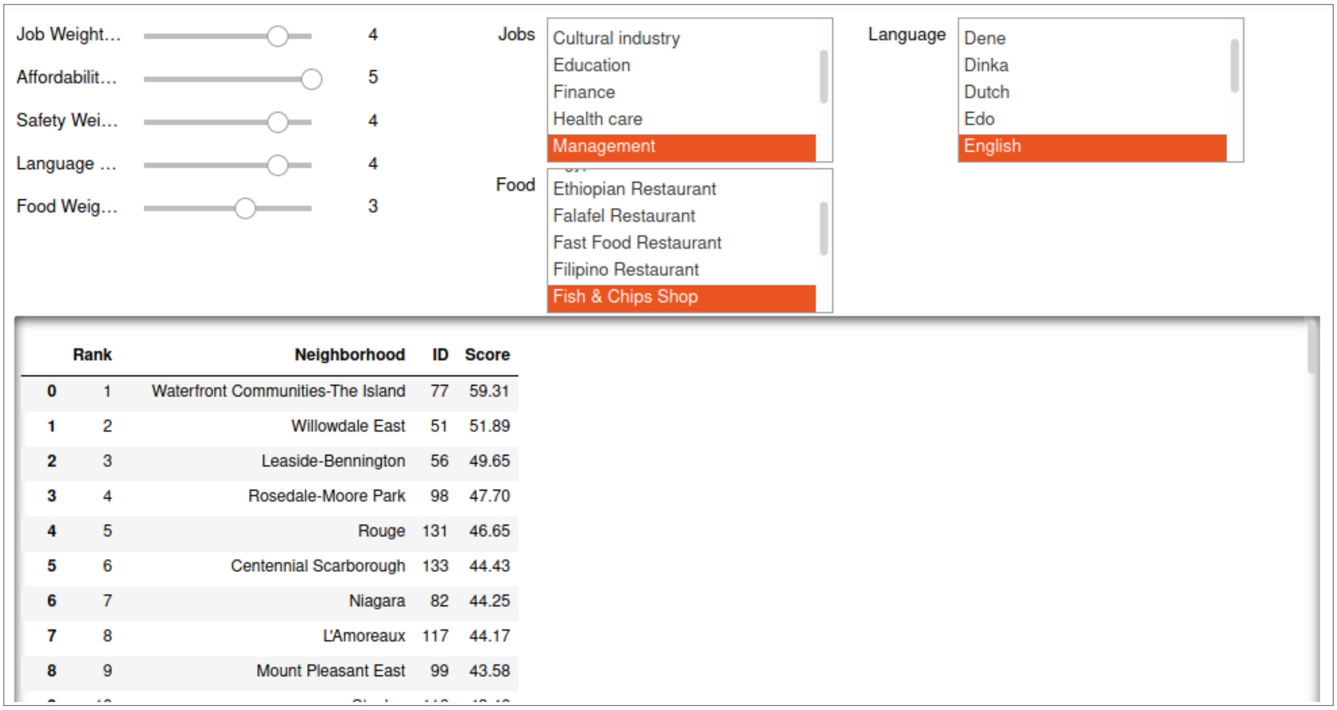
b) Read all data required. We have jobs, affordability, crime rate, language and food data. So, we will rank the neighborhoods based on these 5 factors.

c) Import ipywidgets and numpy libraries.

d) Define get_score function to get the score of each neighborhood based on our choices. Basically this function will sum up the normalized values of our choices and sort them in descending order. This function will use norm_jobs_df, norm_affordability_df, norm_crime_df, norm_language_df and norm_food_df.

e) Set up widgets to filter the results.

f) Use interactive to call the get_score function to rank the neighborhoods by their score. Figure 3.1 shows the result returned by the get_score function.



	Rank	Neighborhood	ID	Score
0	1	Waterfront Communities-The Island	77	59.31
1	2	Willowdale East	51	51.89
2	3	Leaside-Bennington	56	49.65
3	4	Rosedale-Moore Park	98	47.70
4	5	Rouge	131	46.65
5	6	Centennial Scarborough	133	44.43
6	7	Niagara	82	44.25
7	8	L'Amoreaux	117	44.17
8	9	Mount Pleasant East	99	43.58

Figure 3.1: The result returned by the get_score function

I set the default values of job weightage, affordability weightage, safety weightage, language weightage and food weightage to 4, 5, 4, 4 and 3 respectively. I set the default jobs as professional and management, default languages as English and Mandarin, default food as fish & chips shop and pizza place. And the top 1 neighborhood returned by the recommender system is Waterfront Communities-The Island with the score of 59.31%, the higher the score, the higher the chance we will like the neighborhood. If we change the weightage of each factor and choose other types of jobs, languages or food, it will return another neighborhood that suit us the best. Therefore, a Toronto's neighborhoods recommender system is born!

3.2 Data visualization

Now, we will use some attractive plots to visualize the results. We use **Plotly** [14] library to perform the data visualization. Let's return the score of each neighborhood on a choropleth map first.

a) Import requests, geocoder and plotly libraries.

b) Get the toronto boundary data into toronto_geojson.

c) Define a function to visualize the score of each neighborhood on a choropleth map. Basically this function will use the results returned by the get_score function to generate a choropleth map.

d) Use interactive again to show the results. Figure 3.2 shows the choropleth map of Toronto city.

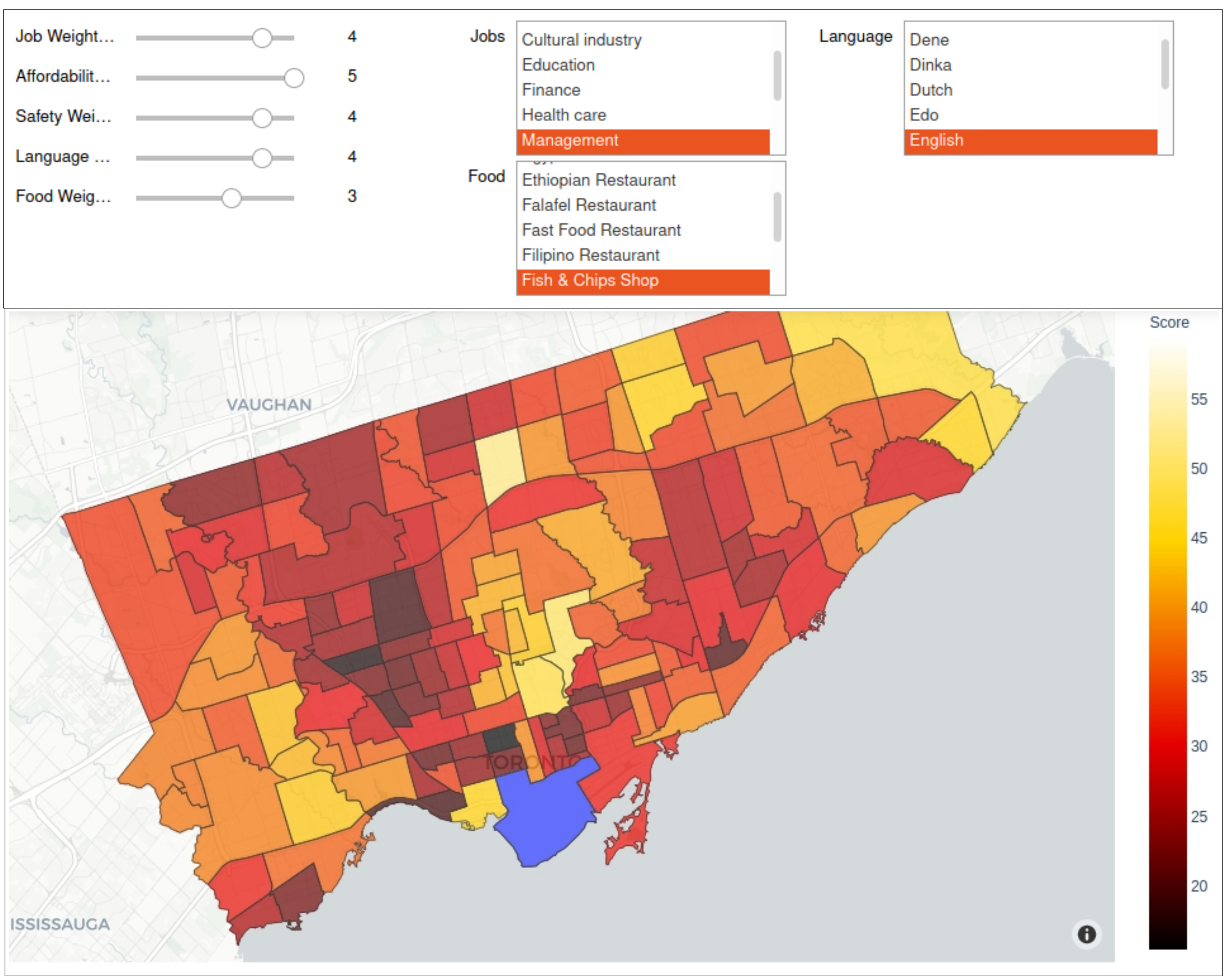


Figure 3.2: The choropleth map of Toronto city

The choropleth map displays the score of each neighborhood with different color scale. And the choropleth map produced by Plotly is highly interactive. As we hover over the neighborhoods, the information about the rank, name and score of each neighborhood will be displayed.

Now, let's visualize the top 5 common jobs, languages and food categories of the top 1 neighborhood returned by the recommender system which is Waterfront Communities-The Island on a sunburst figure.

a) Import plotly library.

b) Set the layout of the Plotly's figures.

c) Use the top5_jobs_df, top5_language_df, top5_food_df, percent_jobs_df, percent_language_df and percent_food_df to create a sunburst figure. Figure 3.3 shows the sunburst figure of Waterfront Communities-The Island.

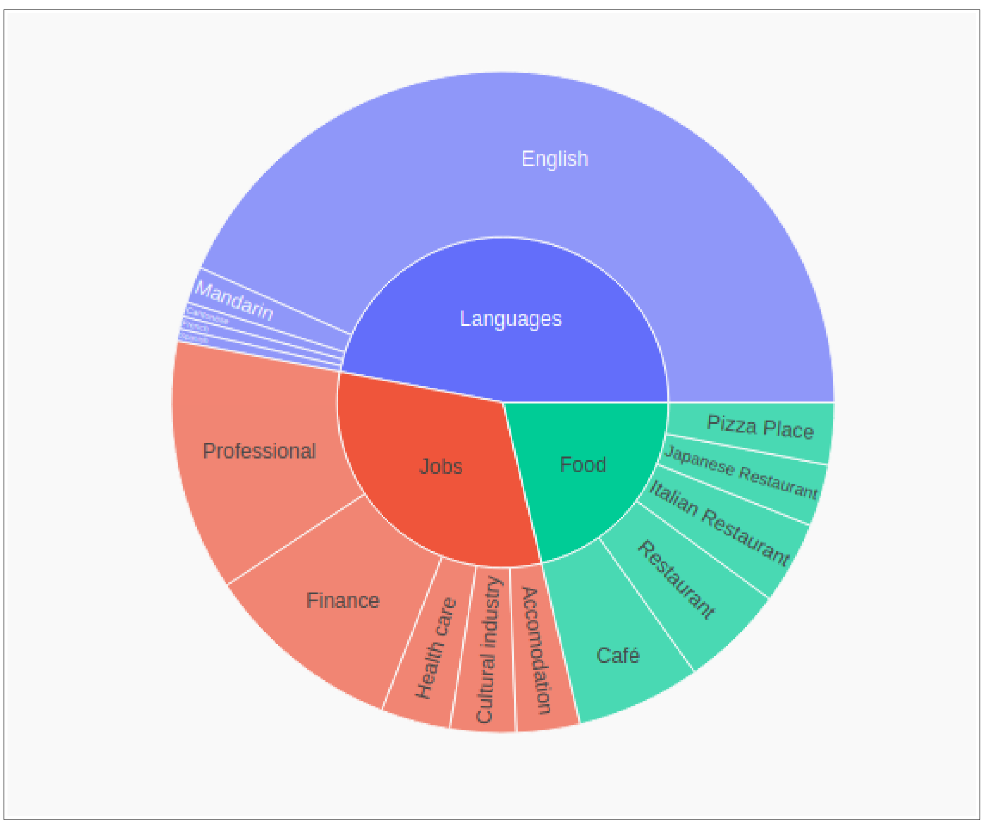


Figure 3.3: The sunburst figure of Waterfront Communities-The Island

The sunburst figure shows the top 5 common jobs, languages and food of the Waterfront Communities-The Island clearly and effectively. As you hover over an item of the figure, it will display the information like its label and percentage. One more advantage of sunburst figure is you can click on the category that you are interested in and it will display only the category for you. For example, if you're interested in the jobs category, you can simply click on the 'Jobs', and the sunburst figure will transform and show you the information of jobs only. Once you're satisfied with it, you can click again the 'Jobs', and it will return to its original state. By comparing our choices with the sunburst figure, it proves that the recommender system has done a pretty good job. Waterfront Communities-The Island has almost all the elements that we want such as professional jobs, English and Mandarin languages, and pizza place.

Now, let's visualize the crime rate and affordability index of Waterfront Communities-The Island on 2 indicators.

a) Compute the average crime rate and affordability index of Toronto city.

b) Use crime_df and affordability_df to create 2 indicators. Figure 3.4 shows the indicators of Waterfront Communities-The Island.

Figure 3.4: The indicators of Waterfront Communities-The Island

The indicator is a simple and effective way to convey numerical information. Green arrow means good result and red arrow means bad result. The affordability index of Waterfront Communities-The Island is 116 which is 43% higher than the average. However, the crime rate of this neighborhood is 1960 per 100,000 people which is 60% higher than the average. These indicators mean that Waterfront Communities-The Island is a highly affordable but unsafe neighborhood compared to others.

3.3 Build an interactive dashboard

In this part, I will convert the recommender system to an interactive dashboard. I use **Dash** [15] library to create the dashboard then deploy the Dash app on **Heroku** [16]. Go to results section to see how the dashboard looks like.

a) Import necessary libraries namely dash, plotly, pandas, numpy and requests.

b) Initialize dash app and read all data required.

c) Set layout or styling of the Plotly's figures.

d) Set the markdown of the dashboard.

e) Define function to visualize the score on a choropleth map.

f) Define function to visualize the top5 jobs, languages and food of selected neighborhood on sunburst figure.

g) Define function to visualize the crime rate and affordability index of selected neighborhood on indicator.

h) Define function to get the score of each neighborhood.

i) Set up initial data and the layout of dashboard.

j) Define callback for click action.

k) Define callback to update the toronto map, sunburst figure and indicator.

l) Run the dash app locally.

m) If everything looks nice, follow **these steps** [17] to deploy Dash app to Heroku.

4. Results

Yo, Figure 4.1 shows the Toronto's Neighborhoods Recommender System! Follow this **link** [18] if you wanna to try it out yourself and find out the neighborhood that suits you the best.

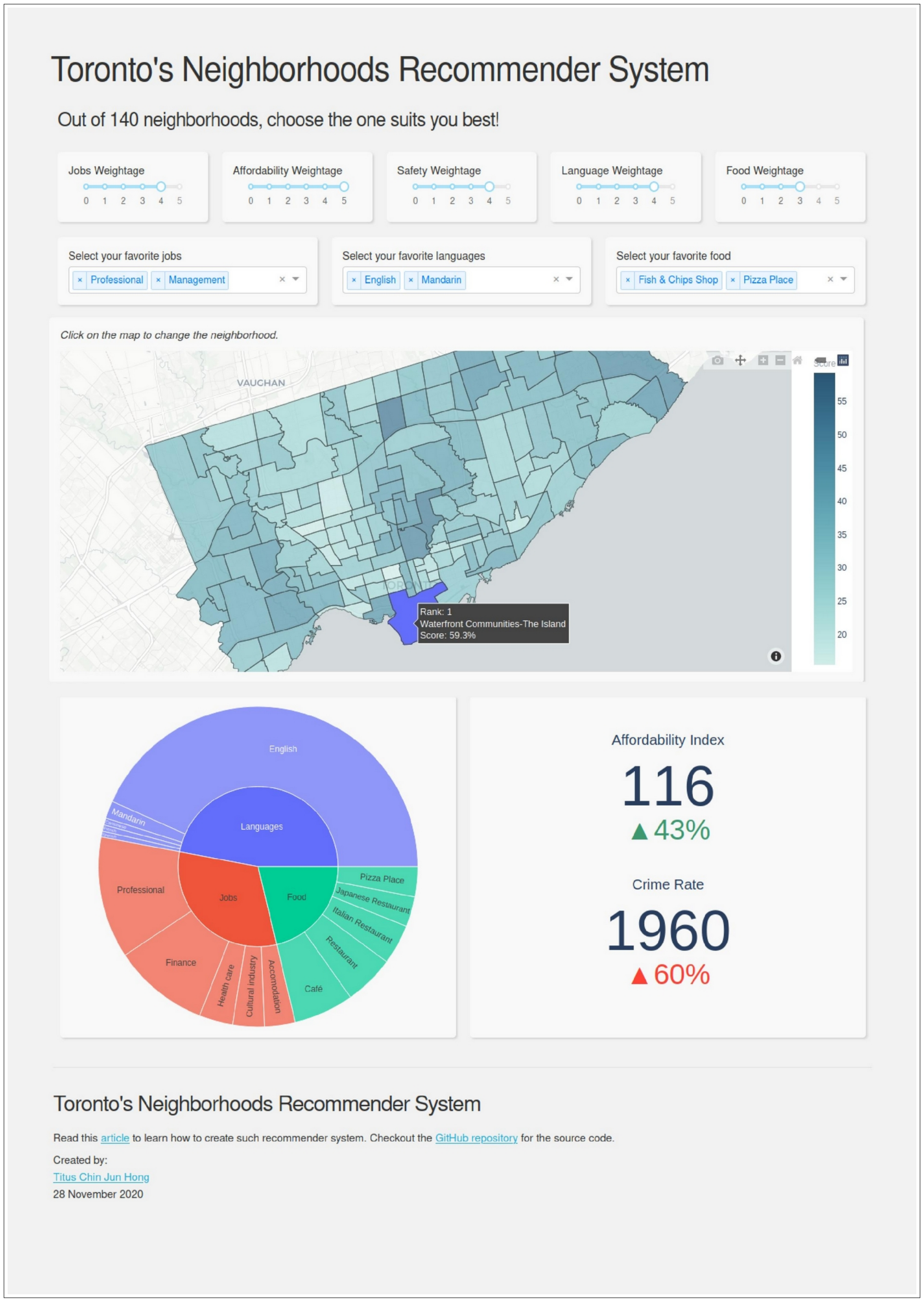


Figure 4.1: The Toronto's Neighborhoods Recommender System

As you can see, the Toronto's neighborhoods recommender system will return the score of each neighborhood based on our choices, and rank them accordingly, then visualize the results on a choropleth map. Besides, the recommender system will also display the top 5 common jobs, languages and food categories for the selected neighborhood on a sunburst figure. The affordability index and crime rate of the selected neighborhood also will be shown. We can set our preferences by using the weightage sliders and selecting our favorite jobs, languages and food, so it's highly personalized! For your information, there are 20 jobs, 127 languages and 88 food categories available for us to choose. If the recommender system returns a very high score result, it means that the neighborhood has met most of our choices. However, if it returns a low score result, don't be sad too, it still shows us the one that suits us the best although we know that not all our choices are met. So, the Toronto's Neighborhoods Recommender System is born! Let's call it TNRS.

5. Discussion

TNRS is very useful for those who wish to migrate to Canada and settle in Toronto city. Out of 140 neighborhoods, it's hard to decide which one suits us the best. We can of course google and see others' reviews about those neighborhoods but it's troublesome and different people have different tastes. That's why I decided to come out with this recommender system to help people choose the neighborhood that suits them the best based on their preferences. It's quite successful, however it still has some limitation. The most obvious one is the long loading time. After setting our preferences or clicking on the map, we have to wait for approximately 10-60 seconds. So, be patient. The reasons behind this is the algorithm to build the recommender system is not optimized, it has poor time complexity, so it takes longer time to compute the scores and return the map, sunburst figure and the indicators. To be honest, I am new to Data Science, so I can't come out with a better algorithm. If possible, I hope someone can look into my python scripts and teach me how to optimize it for a better results. Secondly, the initial loading time is long too. This is because I deployed TNRS on Heroku with a free account. So, after 30 minutes inactive, the app will automatically sleep. If someone attempt to access to the app again, it basically takes about 15-30 seconds to initialize the app. The solution for this is upgrade my Heroku account but I think I will just leave it like that.

6. Conclusion

As a conclusion, I have successfully come out with a Toronto's neighborhoods recommender system (TNRS). I hope it will be helpful for those who love Toronto. Thank you for reading this report! Feel free to read this **article** [19] to learn how to create this recommender system. Checkout the **GitHub repository** [9] for the source code. Access to TNRS via this **link** [18].

7. References

1. K. El-Assal, 'Which cities in Canada attract the most immigrants?', 2020. [Online]. Available: https://www.cicnews.com/2020/02/which-cities-in-canada-attract-the-most-immigrants-0213741.html#. [Accessed: 18- Nov- 2020].
2. M. Niren, '10 Reasons Why Immigrants Choose to Settle in Toronto', 2020. [Online]. Available: https://www.visaplace.com/blog-immigration-law/why-immigrants-settle-in-toronto-heres-10-reasons/. [Accessed: 18- Nov- 2020].
3. City of Toronto, 'About Toronto', 2020. [Online]. Available: https://www.toronto.ca/community-people/moving-to-toronto/about-toronto/. [Accessed: 18- Nov- 2020].
4. City of Toronto, 'About Neighborhood Profiles', 2019. [Online]. Available: https://ckan0.cf.opendata.inter.prod-toronto.ca/download_resource/ef0239b1-832b-4d0b-a1f3-4153e53b189e?format=csv. [Accessed: 18- Nov- 2020].
5. Realosophy, 'Explore Toronto Neighborhoods', 2020. [Online]. Available: https://www.realosophy.com/toronto/neighbourhood-map. [Accessed: 18- Nov- 2020].
6. Toronto Police Service, 'Neighborhood Crime Rates (Boundary File)', 2020. [Online]. Available: https://data.torontopolice.on.ca/datasets/neighbourhood-crime-rates-boundary-file-?geometry=-79.598%2C43.673%2C-79.158%2C43.760&orderBy=OBJECTID&page=6. [Accessed: 18- Nov- 2020].
7. City of Toronto, 'About Neighborhoods', 2020. [Online]. Available: https://ckan0.cf.opendata.inter.prod-toronto.ca/download_resource/a083c865-6d60-4d1d-b6c6-b0c8a85f9c15?format=geojson&projection=4326. [Accessed: 18- Nov- 2020].

8. Foursquare, 'Foursquare Developers', 2020. [Online]. Available: https://developer.foursquare.com/docs/. [Accessed: 18- Nov- 2020].
9. T. Chin, 'Toronto's Neighborhoods Recommender System', 2020. [Online]. Available: https://github.com/titus-chin/Toronto-Neighborhoods-Recommender-System. [Accessed: 18- Nov- 2020].
10. Federal Reserve Bank of San Francisco, 'How is the housing affordability index calculated?', 2003. [Online]. Available: https://www.frbsf.org/education/publications/doctor-econ/2003/december/housing-affordability-index/. [Accessed: 18- Nov- 2020].
11. Ratehub.ca, '1-Year Fixed Mortgage Rates', 2020. [Online]. Available: https://www.ratehub.ca/best-mortgage-rates/1-year/fixed?scenario=purchase&home_price=1000000&down_payment_percent=0.2&downPayment=200000&approximateMortgageAmount=800000&amount=800000&amortization=25&live_in_property=true&pre_approval=false. [Accessed: 18- Nov- 2020].
12. Financial Consumer Agency of Canada, 'How much you need for a down payment', 2020. [Online]. Available: https://www.canada.ca/en/financial-consumer-agency/services/mortgages/down-payment.html. [Accessed: 18- Nov- 2020].
13. Azcalculator.com, 'Crime Rate Formula', 2020. [Online]. Available: https://www.azcalculator.com/formula/crime-rate-calculator.php. [Accessed: 18- Nov- 2020].
14. Plotly, 'Plotly Python Open Source Graphing Library', 2020. [Online]. Available: https://plotly.com/python/. [Accessed: 28- Nov- 2020].
15. Plotly, 'Dash User Guide', 2020. [Online]. Available: https://dash.plotly.com/. [Accessed: 28- Nov- 2020].
16. Heroku. [Online]. Available: https://www.heroku.com/. [Accessed: 28- Nov- 2020].
17. Plotly, 'Deploying Dash Apps', 2020. [Online]. Available: https://dash.plotly.com/deployment. [Accessed: 28- Nov- 2020].
18. T. Chin, 'Toronto's Neighborhoods Recommender System', 2020. [Online]. Available: https://toronto-neighborhoods-app.herokuapp.com. [Accessed: 28- Nov- 2020].
19. T. Chin, 'Toronto's Neighborhoods Recommender System', 2020. [Online]. Available: https://www.linkedin.com/pulse/torontos-neighborhoods-recommender-system-titus-chin/?trackingId=enK3bhKaRlCeV1dh7iHSJQ%3D%3D. [Accessed: 03- Dec- 2020].