Applied Data Science Capstone

The Battle of Neighborhoods

Business Case:

A headhunter and international staff recruiter wants to offer *after recruitment services* for selected candidates by recommending them their favorite neighborhoods in Toronto, Canada.

There is a corresponding Jupyter notebook with the full code which can be downloaded from here:

https://github.com/StefanSiegert/Coursera_Capstone/blob/master/CapstoneWeek4and5.ipynb

There also is a presentation which gives a short impression of the problem to solve, what was done and how were the results, downloadable from here:

https://github.com/StefanSiegert/Coursera_Capstone/blob/master/CapstoneWeek4and5_presentation.pdf

Final report by Stefan Siegert

May 29th, 2020

Table of contents

(1) Introduction of the business problem and who would be interested

This is Kathy. She works as a headhunter and international staff recruiter in Toronto, Canada since more than 10 years. Her business is running very well and her customers highly appreciate her professionalism as well as her fruitful input to their companies' staff structure improvement. Local business association already called her the #1 recruiter in town since she always seems to find the one and only candidate somewhere in the world for any highly specialized position while internal HR departments looked for such new staff for a long time without any success.

However, competition is tough and the market for what she does is limited only to a few business orders per week. So, instead of relaxing and enjoying her achievements, she continuously observes current developments and always tries out to offer anything more than other headhunters do. Since some years, for example, she extended here activity to a business section that she calls 'after recruitment services', where she serves her customer companies with additional data she gained by evaluating the whole recruiting process and comparing it with previous business cases. Additional services she offers to the recruited candidates as well, who very often come from foreign countries to Toronto without any knowledge of this place.

'If a candidate is happy after arrival, his contribution to make his new employer more successful will rise as well', she thinks. Several times she already cooperated with a real estate company to ease the candidate's way to a new home after moving around the world just for this job. Unfortunately, from time to time, there was some dissatisfaction from the candidate's perspective in the past. They got a wonderful apartment with a nice view, but later they recognized that their new neighborhood did not match their preferences very well. It turned out, that the real estate company was very happy about customers who do not have this in-depth knowledge of Toronto's local context and did not care much about anything else than profiting quickly from them.

On Kathy's permanent way of optimizing her business, she now sends out a short questionnaire to her candidates to learn what kind of venues they really prefer to have in their neighborhood instead. Her idea is, to recommend them their perfect neighborhoods so they are better prepared when they look for a new home.

After some internal evaluation she found out that there are four main categories of preferred neighborhoods among her staff candidates:

1) Some hipsters for example want to have coffee shops and sushi places around.
2) Candidates with families prefer parks or playgrounds close to their future home.
3) Sportive candidates need a gym in their neighborhood.
4) Others can mainly be satisfied if they have enough options to go shopping.

She does not only offer this to the candidates but also likes to share it with these companies she got the recruiting orders from. 'You can book a hotel in their preferred

areas if you invite candidates for an interview. So you can make sure that they have a great stay which may convince them even more to work for you in the future.', she says to her clients. They really like this idea and they are confident again and again that she is still the best and most innovative recruiter in town.

Kathy's problem: though she knows her home town very well, it's also no secret to her that trending neighborhoods come and go, gentrification can change streets and surroundings completely in a very short period of time and that her own subjective view on such city developments is not representative enough and of course not very scientific.

So she decides to ask a **data scientist** to help her defining these favorite neighborhoods and to create a map of it for her candidates and the companies who want to hire them.

(2) Data that will be used to solve the problem and source of the data

The general idea is to check all neighborhoods in Toronto and to find out if and how much they match one of the four categories, which scientifically we will call clusters here, mentioned above.

First, it is necessary to define what exactly is a neighborhood. Since Canada has a **postal code system** which is similar to the British one, already representing a high degree of localization, it is decided to call an area with one unique ZIP code a neighborhood for this task. This specific data for Toronto is, fortunately, already available on a Wikipedia page[1].

In a next step, we need to find out the exact position of each of these neighborhoods, defined by **latitude and longitude**. There are several ways to get this data, for example the crowd sourcing project of Openstreetmap offers a so-called 'Overpass API'[2] that could be used for free, Geopy would be another option. For this example, we will simply import a *.csv file[3] that already includes this data and add it to our neighborhoods list.

The data of existing **venues in each neighborhood** can be accessed in our Python code via the Foursquare Places API[4]. We will create a ranking which are the most common venues in these neighborhoods. We will import the Folium library and data provided from that **source to display maps** using Openstreemap of all specified clusters in the end.

Which kind of data will exactly match **our four clusters** mentioned above will be described more detailed in the next chapter, the 'methodology section'.

1 See https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M
2 See https://wiki.openstreetmap.org/wiki/Overpass_API
3 See https://cocl.us/Geospatial_data
4 See https://developer.foursquare.com/places

(3) Methodology section

(a) Import required libraries and create PANDAS dataframe from Wikipedia table

Before we start, we have to import and install some relevant libraries and tools. Please note that not all of them are required for this concrete example here, like parts of geopy or kmeans. However, since this program structure can also used for many other tasks, where exactly these components would be needed, they are still included here.

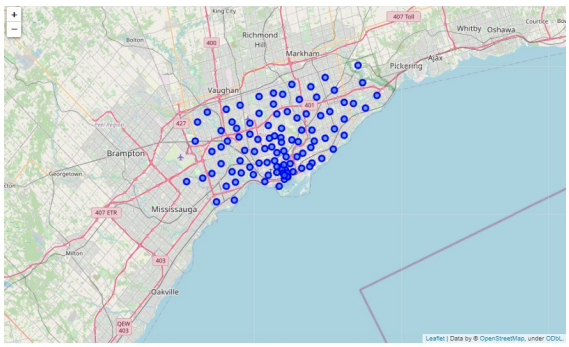
Followed by this, the rough data about postal codes in Toronto will be read from the Wikipedia page and will be converted to a PANDAS dataframe.

The code block ends, like many following code blocks, with the option to show the current state of the dataframe. It is de-activated here by adding a comment sign '#' which can easily be removed if necessary.

(b) Add latitude and longitude to data frame

We add 2 more columns to our df_toronto dataframe ('Latitude' and 'Longitude'), read *.csv file including this data and add it to all rows in the dataframe.

(c) Create a map of Toronto with all neighborhoods



Please note: What is done here is very similar to the example **Segmenting and Clustering Neighborhoods in New York City**. For that reason, many parts of the source code as well as comments and markups were copied and only modified where necessary.

We use geopy library to get the latitude and longitude values of Toronto. Then we will create a map with neighborhoods superimposed on top. The map looks finally like this, neighborhoods are clickable to receive more information.

(d) Connecting to Foursquare API to explore the neighborhoods

Next, we are going to start utilizing the Foursquare API to explore the neighborhoods and segment them. Then, let's create the GET request URL and name our URL url. Finally, the code to run some newly created functions on each neighborhood and create a new

dataframe called toronto_venues is created. Since analyzing takes some time, each single neighborhood will be displayed so the user can see the progress.

(e): Check data as accessed until now

This step is not necessary but makes sense to check if all data was added correctly until now. Let's find out how many unique categories can be curated from all the returned venues and let's check how many venues were returned for each neighborhood.

(f): Analyze each neighborhood and make ranking

This step is very important. Each single neighborhood in our dataframe will be connected to the most common venues in the area. Let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category. Finally, let's print each neighborhood along with the top 5 most common venues that it looks like this.

```
----Garden District, Ryerson----
                       venue  freq
0           Clothing Store  0.09
1              Coffee Shop  0.08
2                Restaurant  0.03
3  Middle Eastern Restaurant  0.03
4           Cosmetics Shop  0.03
```

(g): Transfer this to a new dataframe

Let's put that into a pandas dataframe. First, let's write a function to sort the venues in descending order. Then let's create this new dataframe with the top 10 venues for each neighborhood.

(h): Defining our own clusters and clustering the whole data

First, the grouped venue data must be merged with our df_toronto dataframe. Let's call this new dataframe df_toronto_areastyles.

Then, we have to check each row if it matches our cluster criteria and set the cluster value in the 'Cluster Lables' column. If a neighborhood does not match to any of our clusters, we will remove that row from the dataframe.

Now we have to define exactly if a neighborhood is matching our cluster definition or not. Remember that our four clusters for this topic are:

1) Some hipsters who want to have coffee shops and sushi places around.
2) Candidates with families prefer parks or playgrounds close to their future home.
3) Sportive candidates need a gym in their neighborhood.
4) Others can mainly be satisfied if they have enough options to go shopping.

By iterating through our whole dataframe, we will first convert the current row from the

series type to a list because it's easier to handle for our upcoming tasks. We also create 3 categories of this list: One called 'row_list' with data of all 10 most common venues, 'row_list_top6' with the top 6 ranking common venues and 'row_list_top3' that represent only the top 3 common venues in this neighborhood.

```
for index, row in df_toronto_areastyles.iterrows():
    newcluster = False

    row_list = row.values.tolist()
    row_list_top3 = row_list[0:9]
    row_list_top6 = row_list[0:12]
```

We define our **cluster #1** ('hipster's happiness')[5] if there is a café or a coffee shop ranking in the 3 most common venues around, plus, there must be sushi places or Japanese restaurants in the top 10 ranking. This is the corresponding code to check:

```
    # Check for cluster #01: hipster's happiness
    if (row_list_top3.count('Coffee Shop') == 1)  & (row_list_top3.count('Café') == 1) & ((row_list.count('Sushi Restaurant') ==
1) | (row_list.count('Japanese Restaurant') == 1)):
        df_toronto_areastyles.at[index,'Cluster Labels'] = '0'
        newcluster = True
```

Now let's check for **cluster #2** ('family friendly areas'). For this exercise we simply say that as long as parks and playgrounds are somewhere in the top 10 ranking, criteria will match.

```
    # Check for cluster #02: family friendly areas
    if (row_list.count('Park') == 1)  & (row_list.count('Playground') == 1):
        df_toronto_areastyles.at[index,'Cluster Labels'] = '1'
        newcluster = True
```

Cluster #3 ('workout areas') require in our case gyms or fitness centers ranking somewhere in the top 6 of most common venues in a neighborhood. In Python code, we can check for it in this way:

```
    # Check for cluster #03: workout areas
    if ((row_list_top6.count('Gym') == 1) | (row_list_top6.count('Gym / Fitness Center') == 1)):
        df_toronto_areastyles.at[index,'Cluster Labels'] = '2'
        newcluster = True
```

From the technical point of view, **cluster #4** ('shopping to the max') is slightly different. The main reason is that in the other clusters we could easily count if a venue was included in the ranking or not, and if it was there, the neighborhood matched our criteria. Exploring if there are enough options to go shopping is not that easy because we have to check for any kinds of shops and stores and not just for one kind of venue like 'Park' or 'Gym'.

We solve this by checking if 3 or more most common venues in a neighborhood include one of the the expressions 'Store', 'store', 'Shop', 'shop', 'Mall' or 'Supermarket' in the first 6 top ranking positions. For each iteration we reset a temporary variable a=0, then we iterate through our whole top 6 ranking and increase this value by one if one of these expressions is included. Unfortunately, that's still not all: since 'Coffee Shop' also contains the word 'Shop', we have to make sure that we do not count coffee shops as a conventional

5 Please note that for internal programming reasons it is easier to start our cluster numbering with 0 instead of 1. So our dataframe will include '0' for cluster #1, '1' for cluster #2 and so on, which has no influence on the results of course.

shopping venue. So, our neighborhood is only matching cluster #4 if there are 3 or more shops, malls, supermarkets or stores without coffee shops in a neighborhood's top 6 ranking of most common venues.

The final Python code for this:

```
    # Check for cluster #04: shopping to the max
    a = 0
    for b in range(0,10):
        venue = str(row_list_top6[b])
        if ('Store' in venue) | ('Shop' in venue) | ('shop' in venue) | ('Mall' in venue) | ('store' in venue) | ('Supermarket'
in venue):
            if venue != 'Coffee Shop':
                a = a+1
    if a>2:
        df_toronto_areastyles.at[index,'Cluster Labels'] = '3'
        newcluster = True
```

If any of the clusters 1 to 4 was matching, we set the 'newcluster' boolean to True. If no cluster was matching until the end of the current iteration, we will remove this neighborhood from our dataframe because it's not needed anymore for our work. Finally, let's show the left over dataframe in the end.

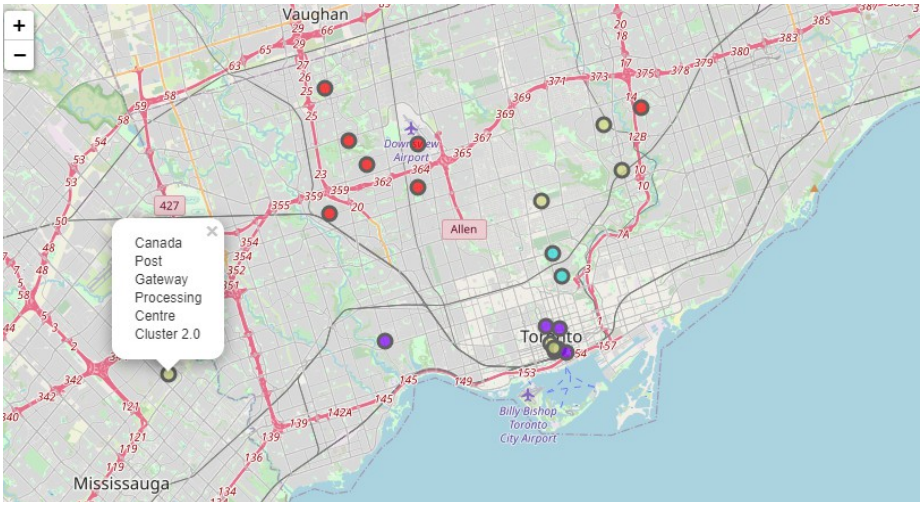
```
    if newcluster == False:
        df_toronto_areastyles = df_toronto_areastyles.drop(index, axis=0)

df_toronto_areastyles
```

In '*Appendix I*' you can see how our final results dataframe with all neighborhoods that were matching our cluster criteria and the cluster number in the 'Cluster Labels' column look like[6].

(i): Creating the map with our clustered neighborhoods

Finally, let's visualize the resulting clusters in a map. By clicking a clustered neighborhood, you will get additional information including our cluster label number.



6 Please note that the columns 'PostalCode', 'Borough', 'Latitude' and 'Longitude' were removed here to allow better overview. If you use the Jupyter notebook instead you will see these columns and the details included in it as well.

(4) Results section including discussion of results.

Kathy is very happy, because with help of the **data scientist** she knows the neighborhoods in Toronto where her recruited candidates would prefer to live very well now. A map is shown on the previous page, a full list can be seen in *'Appendix I'* at the end of this report.

1. Preferred areas of her hipster candidates (cluster #1) are Garden District, Ryerson, Central Bay Street, Toronto Dominion Centre, Design Exchange, Runnymede and Swansea.

2. Family candidates (cluster #2) should move to Moore Park, Summerhill East, Milliken, Agincourt North, Steeles East, L'Amo or Rosedale.

3. For those who love doing workout and gym (cluster #3) she can recommend Don Mills, Richmond, Adelaide, King, Commerce Court, Victoria Hotel, Davisville North, Canada Post Gateway Processing Centre, First Canadian Place or Underground city.

4. Finally, for those who prefer shopping areas around their home (cluster #4), neighborhoods Weston, Downsview, Lawrence Manor, Lawrence Heights or Parkwoods should be their first choice.

A great thing about this solution is that it can be updated easily and as often as required. Neighborhoods are in a dynamic and continuous process of change, new shops or restaurants are opening, others are closing, so it is no surprise that from time to time also different neighborhoods will be preferred by her candidates. To be able to get always latest data is a big advantage of this data science solution.

(5) Observations noted and any recommendations based on the results.

Though not having any real life personal local knowledge of Toronto, Canada, it seems that all the **data science worked surprisingly well** for this example.

A view on the map shows that family friendly neighborhoods are really surrounded by green areas, and most coffee shop neighborhoods are located in downtown. The shopping cluster includes results in neighborhoods that are mainly located near a big highway outside the city, which seems realistic because these are normally areas where shopping malls and other big stores are built.

Only the cluster for sportive candidates cannot be checked on a map, they are in different places of the town, but if the Foursquare data was accurate, they might deliver good results anyway.

However, **concrete definition of the clusters** leaves still room for improvement of course. It might make sense to prioritize specific venues more than just with this 'Top-3, Top-6, Top-10' ranking. For sportive candidates it might also make sense to check for pools, parks, stadiums, yoga venues or dancing studios in their neighborhood as well. Families would also be interested in schools or kindergartens and hipsters might prefer cocktail bars or small stores with crazy stuff, so refinement of the clusters should deliver even better results in the end.

Another observation is that defining a neighborhood just by a radius of 500 meters around the **latitude and longitude values of a postal code** is not always very practical. Some neighborhoods are bigger than others so maybe relevant data will not be included in the results. Or, the other way around, in downtown area for example it happens from time to time that centers of different neighborhoods are less than 1,000 meters away from each other and results are overlapping.

(6) Conclusion section

Though definition of the clusters was done in a very easy and not too scientific way, it turned out that the used model delivered highly detailed results.

It would be good to do some more refinements to the cluster criteria as mentioned in the previous chapter, also using Foursquare as the only source of data might not be the best way to get the right results.

However, the business case matched well to this work, so I think it is a very good point to start from for doing similar analyzing of similar cases in other towns of the world.

Appendix I: Clustered neighborhoods

Here you can see how our final results dataframe with all neighborhoods that were matching our cluster criteria and the cluster number in the 'Cluster Labels' column look like[7].

	Neighborhood	Cluster Labels	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
0	Parkwoods	3	Park	Convenience Store	Food & Drink Shop	Women's Store	Doner Restaurant	Dim Sum Restaurant	Diner	Discount Store	Distribution Center	Dog Run
3	Lawrence Manor, Lawrence Heights	3	Clothing Store	Accessories Store	Women's Store	Gift Shop	Boutique	Miscellaneous Shop	Event Space	Coffee Shop	Furniture / Home Store	Vietnamese Restaurant
7	Don Mills	2	Beer Store	Gym	Restaurant	Japanese Restaurant	Sporting Goods Shop	Asian Restaurant	Coffee Shop	Café	Dim Sum Restaurant	Italian Restaurant
9	Garden District, Ryerson	0	Clothing Store	Coffee Shop	Café	Japanese Restaurant	Cosmetics Shop	Restaurant	Bubble Tea Shop	Italian Restaurant	Middle Eastern Restaurant	Theater
13	Don Mills	2	Beer Store	Gym	Restaurant	Japanese Restaurant	Sporting Goods Shop	Asian Restaurant	Coffee Shop	Café	Dim Sum Restaurant	Italian Restaurant
24	Central Bay Street	0	Coffee Shop	Café	Italian Restaurant	Sandwich Place	Japanese Restaurant	Bubble Tea Shop	Salad Place	Ice Cream Shop	Burger Joint	Thai Restaurant
30	Richmond, Adelaide, King	2	Coffee Shop	Café	Restaurant	Clothing Store	Gym	Deli / Bodega	Thai Restaurant	Hotel	Sushi Restaurant	Concert Hall
40	Downsview	3	Grocery Store	Park	Liquor Store	Shopping Mall	Food Truck	Discount Store	Hotel	Athletics & Sports	Baseball Field	Bank
42	Toronto Dominion Centre, Design Exchange	0	Coffee Shop	Café	Hotel	American Restaurant	Italian Restaurant	Japanese Restaurant	Salad Place	Seafood Restaurant	Restaurant	Deli / Bodega
46	Downsview	3	Grocery Store	Park	Liquor Store	Shopping Mall	Food Truck	Discount Store	Hotel	Athletics & Sports	Baseball Field	Bank
48	Commerce Court, Victoria Hotel	2	Coffee Shop	Café	Restaurant	Hotel	American Restaurant	Gym	Japanese Restaurant	Seafood Restaurant	Italian Restaurant	Deli / Bodega
53	Downsview	3	Grocery Store	Park	Liquor Store	Shopping Mall	Food Truck	Discount Store	Hotel	Athletics & Sports	Baseball Field	Bank
60	Downsview	3	Grocery Store	Park	Liquor Store	Shopping Mall	Food Truck	Discount Store	Hotel	Athletics & Sports	Baseball Field	Bank
64	Weston	3	Park	Convenience Store	Women's Store	Donut Shop	Dim Sum Restaurant	Diner	Discount Store	Distribution Center	Dog Run	Doner Restaurant
67	Davisville North	2	Department Store	Gym	Park	Sandwich Place	Breakfast Spot	Hotel	Food & Drink Shop	Dog Run	Distribution Center	Dim Sum Restaurant
76	Canada Post Gateway Processing Centre	2	Hotel	Coffee Shop	Gym	American Restaurant	Intersection	Sandwich Place	Middle Eastern Restaurant	Fried Chicken Joint	Burrito Place	Mediterranean Restaurant
81	Runnymede, Swansea	0	Café	Pizza Place	Coffee Shop	Sushi Restaurant	Restaurant	Pub	Italian Restaurant	Yoga Studio	Diner	Smoothie Shop
83	Moore Park, Summerhill East	1	Park	Playground	Tennis Court	Summer Camp	Women's Store	Distribution Center	Dessert Shop	Dim Sum Restaurant	Diner	Discount Store
85	Milliken, Agincourt North, Steeles East, L'Amo...	1	Playground	Park	Doner Restaurant	Dessert Shop	Dim Sum Restaurant	Diner	Discount Store	Distribution Center	Dog Run	Donut Shop
91	Rosedale	1	Park	Playground	Trail	Women's Store	Doner Restaurant	Dessert Shop	Dim Sum Restaurant	Diner	Discount Store	Distribution Center
92	Stn A PO Boxes	0	Coffee Shop	Café	Seafood Restaurant	Restaurant	Cocktail Bar	Beer Bar	Japanese Restaurant	Italian Restaurant	Hotel	Creperie
97	First Canadian Place, Underground city	2	Coffee Shop	Café	Japanese Restaurant	Hotel	Gym	Restaurant	Seafood Restaurant	Salad Place	Steakhouse	Deli / Bodega

7 Please note that the columns 'PostalCode', 'Borough', 'Latitude' and 'Longitude' were removed here to allow better overview. If you use the Jupyter notebook instead you will see these columns and the details included in it.