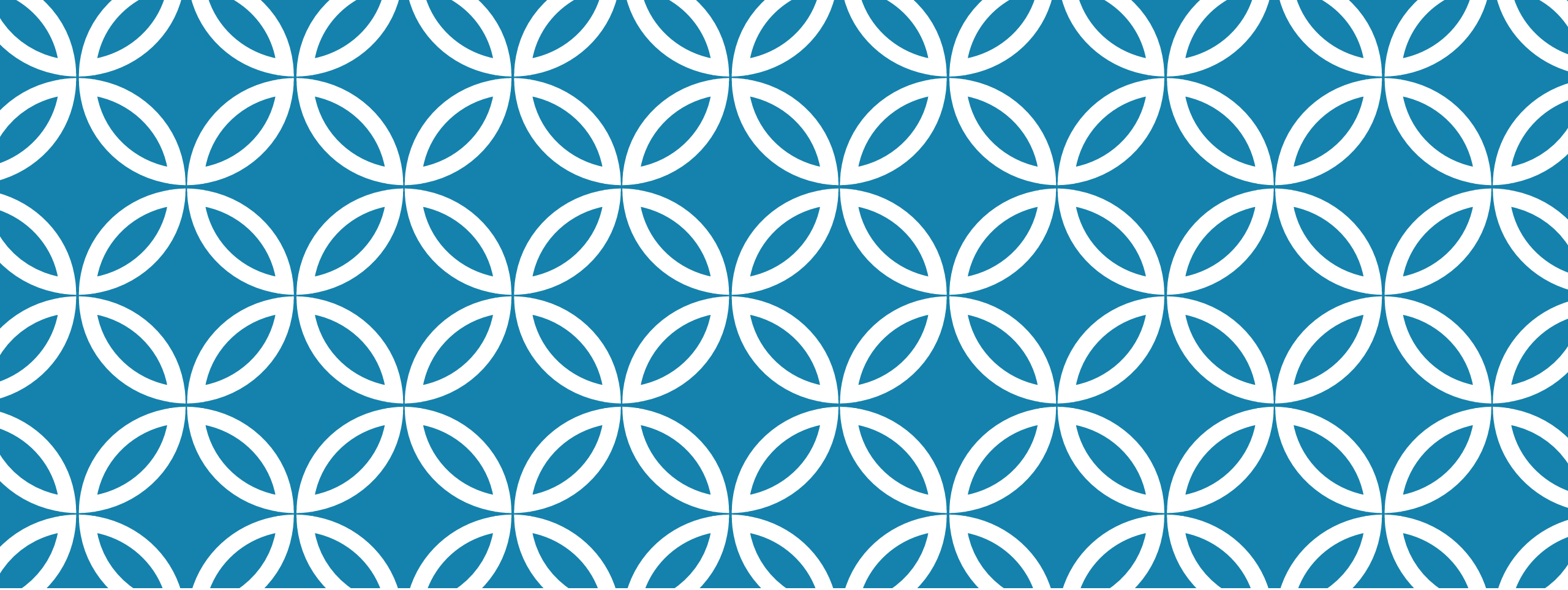


FINDING FAMILIAR (SIMILAR) NEIGHBORHOODS

Mohamed Nadhir DAOUD
04/08/2019

INTRODUCTION

Let's imagine someone who is moving to a new city, among other criteria, he may be interested in finding the most similar (familiar) neighborhood to his actual one.

This project is about finding, in a new city, the neighborhoods most similar to the user actual neighborhood, based on a venue categories search.

DATA

In the Notebook, and in ordered to illustrate how the app would work, I used New York city and Toronto neighborhood data. Both data could be found as csv files on my github repo:

https://github.com/Nadhir10/The_Battle_of_Neighborhoods/tree/master/Data

For the venue categories search we will use the foursquare API.

METHODOLOGY

Euclidian distance

Taking a random neighborhood from **New York** and finding similar neighborhoods in **Toronto**

Clustering

APPROACH 1: EUCLIDIAN DISTANCE

Having tables of the frequency of each venue category for a given neighborhood can be seen as a multidimensional vector representing the neighborhood where every dimension is a venue category.

From that point of view, we could compute the distance between each neighborhood and the user actual neighborhood.

The most similar neighborhoods would be those with the lowest distance to the user neighborhood

APPROACH 2: CLUSTERING

In this approach we would add the user's neighborhood to the list of the other city neighborhoods.

After performing a clustering, The most similar neighborhoods would be those with the same cluster label as the actual neighborhood.

RESULTS

Approach 1 results :

Neighborhood	similarity	Nb of similar categories	Similar categories
Ryerson, Garden District	0.229408	2	[Pizza Place, Fast Food Restaurant]
Chinatown, Grange Park, Kensington Market	0.229459	0	[]
Church and Wellesley	0.236928	0	[]
St. James Town	0.242747	0	[]
Cabbagetown, St. James Town	0.243074	1	[Pizza Place]
Adelaide, King, Richmond	0.243694	0	[]
Harbourfront East, Toronto Islands, Union Station	0.244862	1	[Pizza Place]
Stn A PO Boxes 25 The Esplanade	0.245605	1	[Fast Food Restaurant]
Commerce Court, Victoria Hotel	0.254140	0	[]
Fairview, Henry Farm, Oriole	0.254142	1	[Fast Food Restaurant]

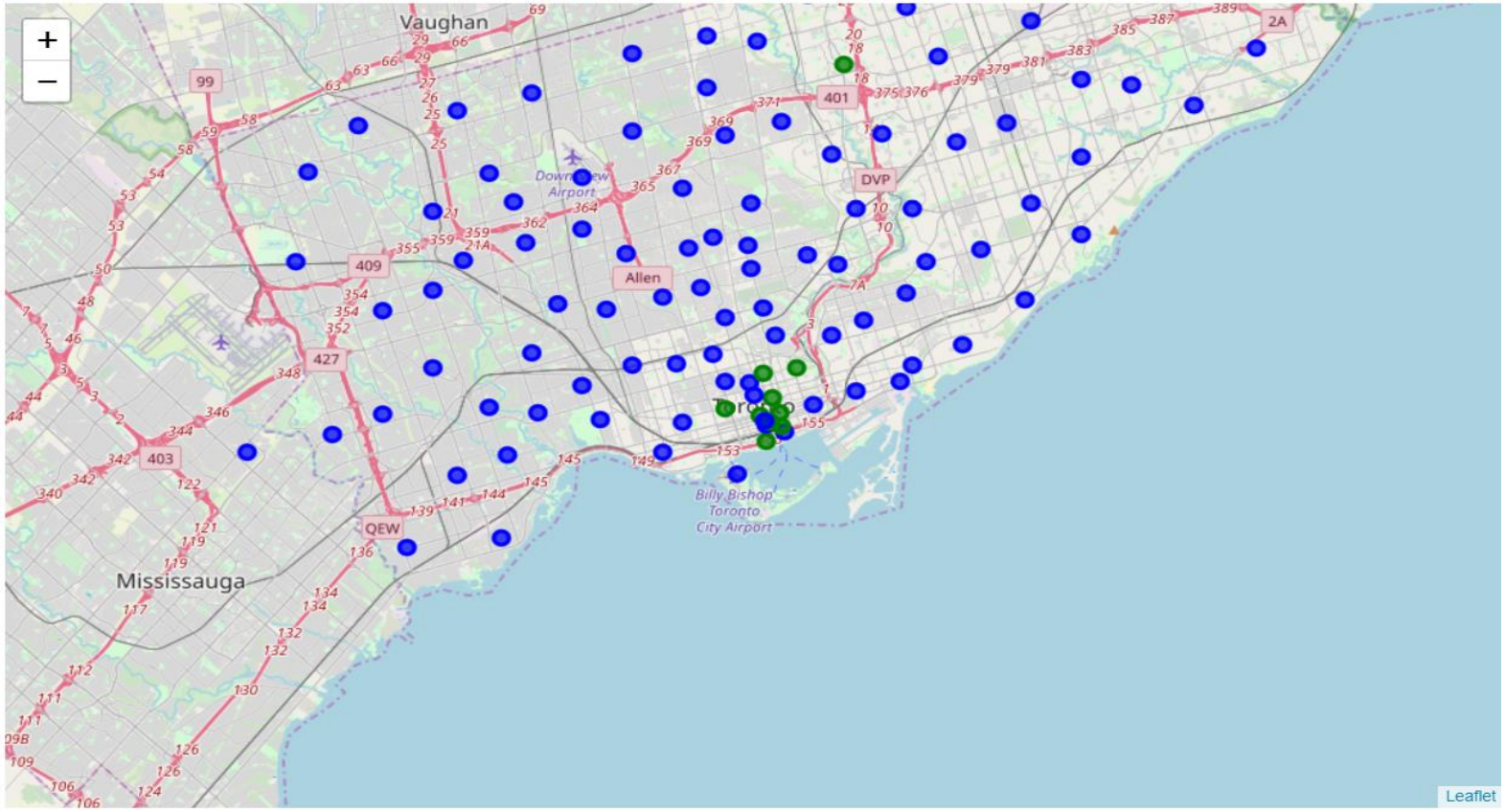
RESULTS

Approach 2 results :

Neighborhood	nb similar categories	Similar categories
Chinatown, Grange Park, Kensington Market	0	[]
Dovercourt Village, Dufferin	2	[Fast Food Restaurant, Pharmacy]
Guildwood, Morningside, West Hill	1	[Pizza Place]
Harbord, University of Toronto	0	[]
High Park, The Junction South	2	[Fried Chicken Joint, Fast Food Restaurant]
Humber Bay Shores, Mimico South, New Toronto	4	[Fried Chicken Joint, Fast Food Restaurant, Pi...
Kingsway Park South West, Mimico NW, The Queen...	1	[Fast Food Restaurant]

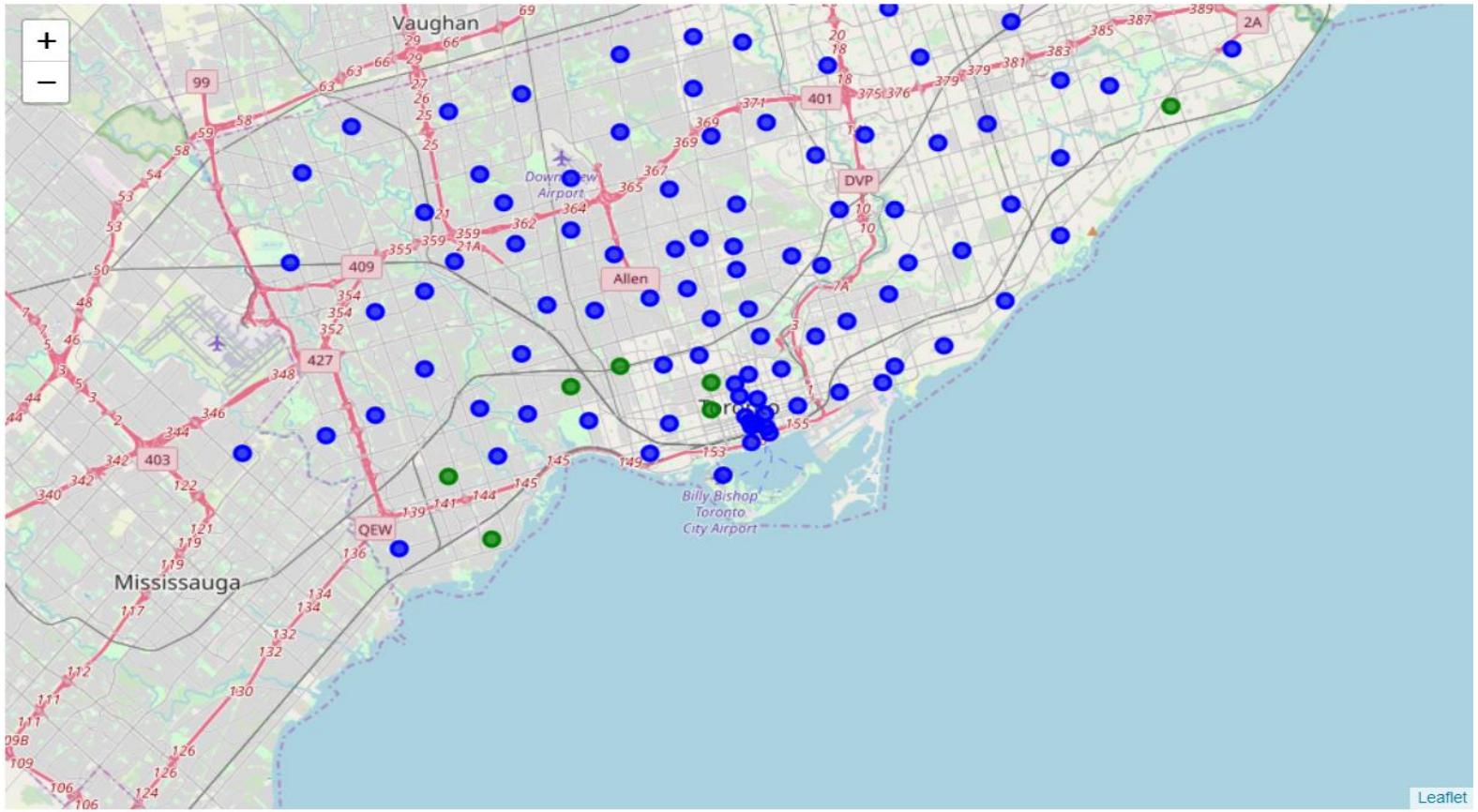
RESULTS

Approach 1 : map of similar neighborhoods in Toronto :



RESULTS

Approach 2 : map of similar neighborhoods in Toronto :



DISCUSSION

Surprisingly the two approaches show different results, and there is only one neighborhood that appears in the results of both approaches (Chinatown, Grange Park, Kensington Market).

Both approaches show neighborhoods that doesn't share enough of their most common venue categories,

The clustering technique shows better results, some of the found neighborhoods have 4 out of 10 most common venue categories,

The choice of the initial number of clusters to start with influences heavily the results,

DISCUSSION

In the other hand, with every new neighborhood tested :

In Approach 1 : We will always find similar neighborhoods, however, the measured distances can vary from very low to very large.

In Approach 2: This choice of the initial number of clusters has to be changed manually.

CONCLUSION

The first approach (Euclidian distance) has the advantage of being more systematic but fails to find as good results as the clustering.

The problem of the clustering is the initial number of clusters to use, it has to be picked manually for each new neighborhood.

Despite this issue, The clustering has shown more reliable and accurate results, therefore I prefer to use the **clustering**.