Which American City is Most Like TORONTO?

Christian J. Harris

January 16, 2021

INTRODUCTION. For the adventurous type of person that wants to explore elsewhere yet still have some sense of familiarity, they may be interested in moving to a city that's similar to where they're from. Case in point, John Doe lives a great life in Toronto but has a lucrative opportunity to move to one of 4 major US cities for the company he works for. The job itself would be much the same but he’s looking forward to a lifestyle outside of work that complements how he already lives. This project seeks to algorithmically determine which of the 4 cities – New York, Chicago, Los Angeles, or Miami – is most like Toronto; or more granularly, are certain neighborhoods in one of these cities much like Toronto as a whole? While we all have our preconceived notions of each city's "flavor," the right data may help reveal lifestyle similarities in small pockets of the US that the soon-to-be-transplant may have not considered.

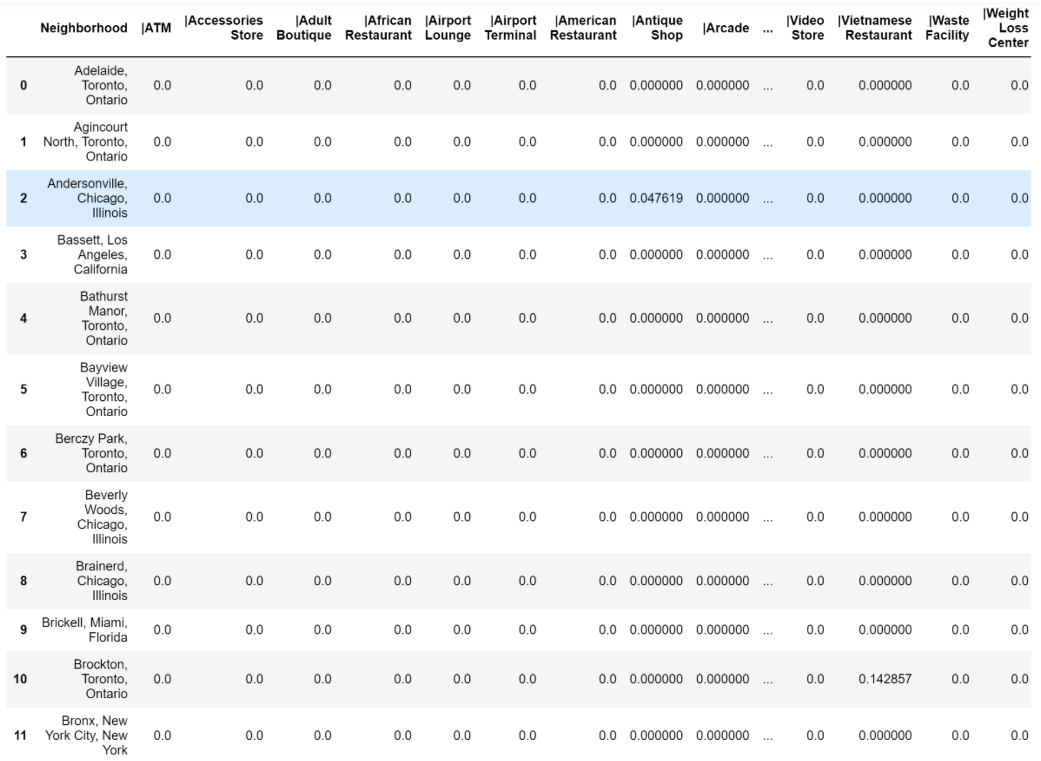
Figure 1. Toronto, Ontario, Canada skyline at dusk.

DATA. All code for analysis was written in Python and processed via a Jupyter Notebook Python kernel. The Python package Pandas was used to scrape tables from a variety of websites for neighborhood names in each of the cities delineated above. Postal Codes were ignored with the logic that neighborhoods provided both greater resolution and increased interpretability. Toronto’s and Miami’s neighborhood listings were gathered from their respective Wikipedia pages, New York City’s neighborhoods were gathered from the New York Department of Health, Chicago’s neighborhoods were gathered from The Chicago Tribune, and Los Angeles’ neighborhoods were gathered from The LA Almanac. The geospatial coordinates of each neighborhood was retrieved via Nominatim, where available; and Foursquare API was used to obtain the venues, and their corresponding venue categories, within a 500-meter vicinity of each neighborhood, when available.

The feature-richness of the dataset consists of the 278 venue categories returned by Foursquare. While features related to Crime Data, Cost-Of-Living, Quality of Education, etcetera would aid our John Doe into a more well-rounded moving decision, the number of features are already more substantial than the number of observations available. Due to the number of cities assessed, various efforts were made to the retrieve the most amount of data possible without causing Nominatim and Foursquare to timeout prematurely. Persistence in pinging servers for repeated unreturned queries was kept to a minimum, given previous failed attempts; each city was limited to no more than 50 neighborhoods, sampled at random, representative of its city. Repeat builds of the final form of the dataset are each unique but yield very similar results; only one such build was used throughout this report.

METHODOLOGY. Regardless of the characteristics used, assessing how alike multiple cities are fits neatly within the realm of an *Unsupervised Clustering* problem. If characterized perfectly enough, them the algorithm would mostly predict a unique cluster for each city, with certain neighborhoods in a different city within the same cluster as Toronto's predominant cluster. In this instance, while 278 venue categories is a lot, it is effectively the only criteria used for clustering. *K-Means Clustering* was the primary means of assessment, followed by *DBSCAN* and *tSNE Clustering* to corroborate the results of K-Means; particularly, DBSCAN was used in the hopes of gleaning more robust cluster predictions than K-Means. *Cosine Similarity* was also used as a single metric to quantify the corroboration visualized by the three Clustering methods.

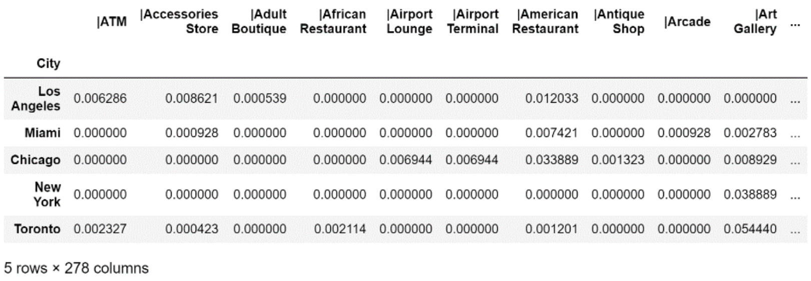
Each row in Figure 2 below is a very sparse, 278-dimensional vector; intuition of the dataset's structural shape is nearly impossible without visualization, hence the use of tSNE to reduce these dimensions down to R2. The final sample of neighborhoods from all 5 cities were concatenated into a single datframe to ensure that clustering labels were consistent and comparable. The venue categories, returned by Foursquare across each neighborhood in this all-cities dataframe, were one-hot encoded and adjusted for their frequency within each neighborhood.



	Neighborhood	|ATM	|Accessories Store	|Adult Boutique	|African Restaurant	|Airport Lounge	|Airport Terminal	|American Restaurant	|Antique Shop	|Arcade	...	|Video Store	|Vietnamese Restaurant	|Waste Facility	|Weight Loss Center
0	Adelaide, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
1	Agincourt North, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
2	Andersonville, Chicago, Illinois	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.047619	0.000000	...	0.0	0.000000	0.0	0.0
3	Bassett, Los Angeles, California	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
4	Bathurst Manor, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
5	Bayview Village, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
6	Berczy Park, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
7	Beverly Woods, Chicago, Illinois	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
8	Brainerd, Chicago, Illinois	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
9	Brickell, Miami, Florida	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0
10	Brockton, Toronto, Ontario	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.142857	0.0	0.0
11	Bronx, New York City, New York	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.000000	...	0.0	0.000000	0.0	0.0

Figure 2. The all-cities dataframe of the first 12 neighborhoods; notice that more than one city is scattered throughout the dataset. The multitude of zeroes represent that particular venue type not being present in that particular neighborhood, while the highlighted neighborhood Andersonville shows that 4.76% of the venues in its vicinity are Antique Shops, and Brockton has Vietnamese Restaurants at 14.29%.

Datapoints (i.e., neighborhoods) in the table shown were clustered via KMeans yet plotted geospatially via the Python package Folium, thereby visualizing venue-similarity across cities. The same was done separately for DBSCAN Clustering. For *Cosine Similarity*, the above table was grouped by city, calculated via the means of each venue category frequency (see Figure 3a below). In other words, a single vector was created for each city that essentially summarized the venue fingerprint of each city as a whole. From here, the Cosine Similarity between each city and Toronto was calculated in Python "manually" via the equation seen in Figure 3b.



	|ATM	|Accessories Store	|Adult Boutique	|African Restaurant	|Airport Lounge	|Airport Terminal	|American Restaurant	|Antique Shop	|Arcade	|Art Gallery	...
City											
Los Angeles	0.006286	0.008621	0.000539	0.000000	0.000000	0.000000	0.012033	0.000000	0.000000	0.000000	...
Miami	0.000000	0.000928	0.000000	0.000000	0.000000	0.000000	0.007421	0.000000	0.000928	0.002783	...
Chicago	0.000000	0.000000	0.000000	0.000000	0.006944	0.006944	0.033889	0.001323	0.000000	0.008929	...
New York	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.038889	...
Toronto	0.002327	0.000423	0.000000	0.002114	0.000000	0.000000	0.001201	0.000000	0.000000	0.054440	...

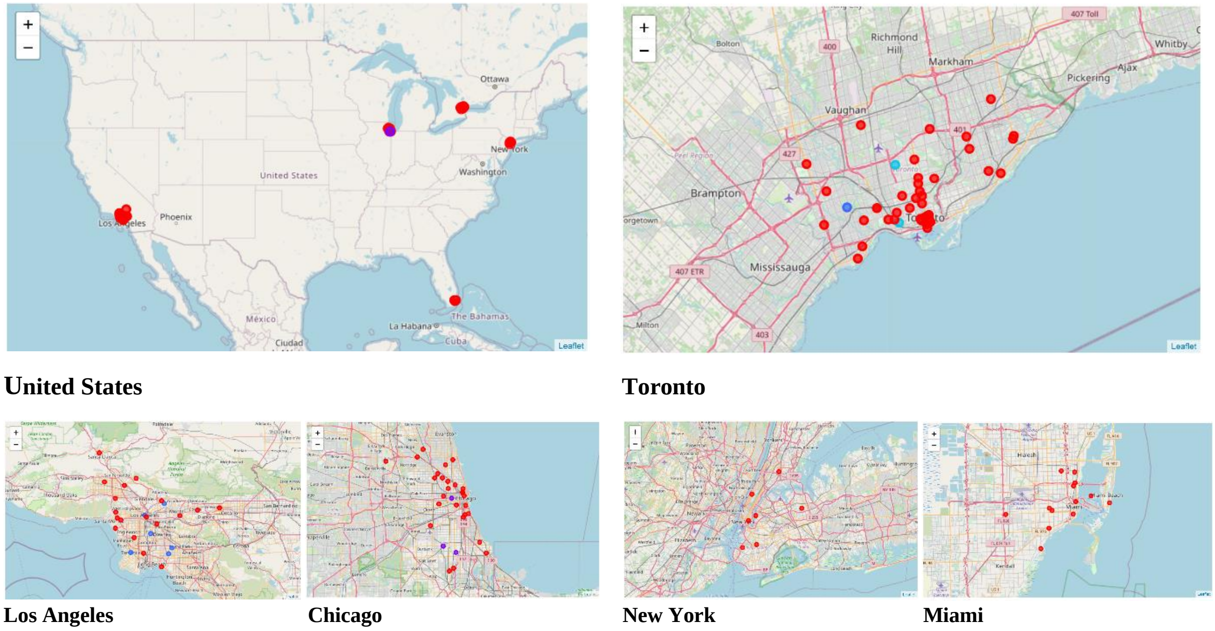
5 rows × 278 columns

$$\frac{\mathbf{A}\cdot\mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

Figure 3. The all-cities dataframe *grouped by city* (left) and the Similarity function used afterwards to compare (right).

RESULTS. Following K-Means Clustering, it can be seen in Figure 4 below that nearly all neighborhoods are encompassed in the same cluster, despite the code calling for a total of 4 clusters. In the zoomed out versiosn of the US map, every city but Chicago appears to be indiscernible from one another. Zooming into each city helps to reveal that there are select neighborhoods that belong to a different cluster.

K-Means (spherical) Clustering, plotted Geospatially



United States

Toronto

Los Angeles

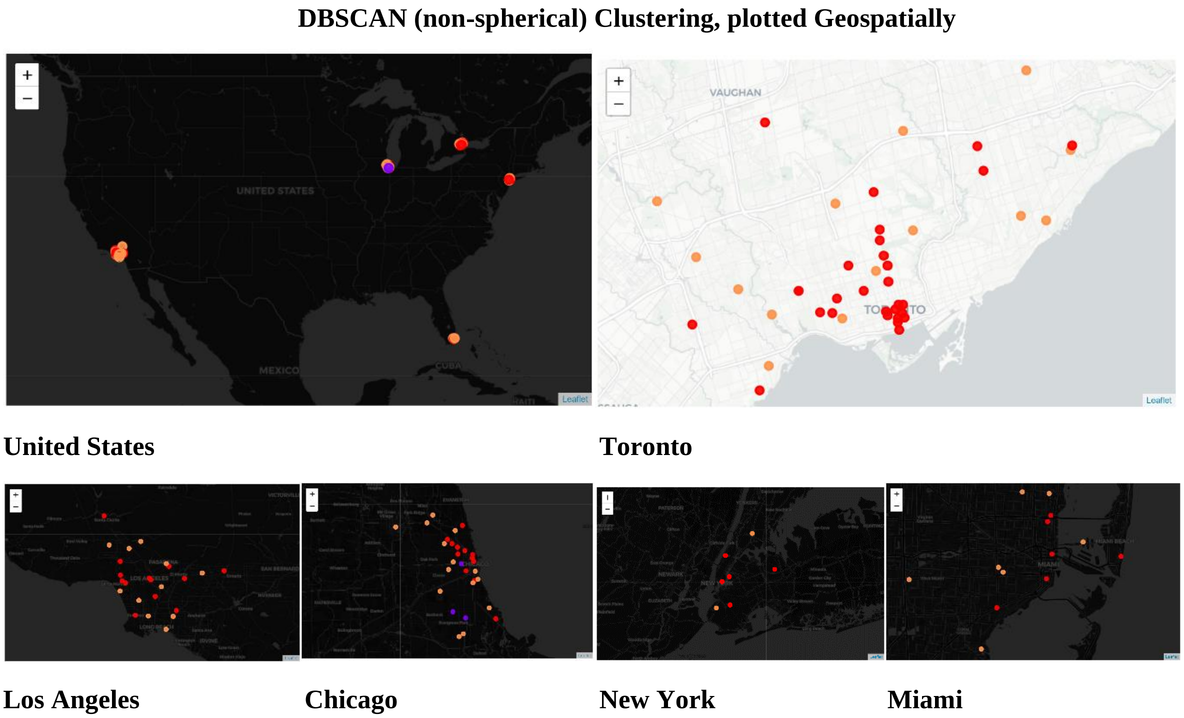
Chicago

New York

Miami

Figure 4. The result of K-Means Clustering shows an overwhelming majority of neighborhoods in a single cluster. As will be discussed in the Discussion section that follows, this is likely due to a combination of Data Sparsity, more dimensions than observations, and/or most neighborhoods simply having a rather similar venue fingerprint, categorically, given no further data.

While we were not necessarily expecting the perfect result of each city belonging to its own cluster, it would've at least been nice to see some level of regionality to the labels within each city. We all know first-hand that not all neighborhoods are created equal the way this K-Means Clustering, coupled with Folium plotting, implies. One thing that comes to mind when seeing this much label imbalance is that, perhaps, the datapoints' structure as a collective is inherently non-spherical. It may be the case that Toronto-like clusters and Chicago-like clusters, etcetera, are present and distinct yet so irregularly shaped and intertwined that K-Means could only encompass them spherically. DBSCAN was performed in the hopes of unfurling these irregular shapes into open view, but the results are as follows:


DBSCAN (non-spherical) Clustering, plotted Geospatially

United States **Toronto**

Los Angeles **Chicago** **New York** **Miami**

Figure 5. The result of K-Means Clustering shows an overwhelming majority of neighborhoods in a single cluster. As will be discussed in the Discussion section that follows, this is likely due to a combination of Data Sparsity, more dimensions than observations, and/or most neighborhoods simply having a rather similar venue fingerprint, categorically, given no further data.

"Progress!," you say, right? Instead of a single predominant cluster, we now have two predominant clusters! While the red dots do represent neighborhoods belonging to a legitimate cluster, the orange dots represent neighborhoods the DBSCAN algorithm classifies as outliers. Despite taking non-spherical data into account, we still essentially assess that most neighborhoods throughout the entire map are much the same. Adjusting the DBSCAN epsilon sensitivity and sample minimum [for each cluster] does yield alternative arrangements of cluster labels, but that all range from almost all red to almost all orange; I discovered no arrangement in which another legitimate cluster had any more than a couple of neighborhoods.

So why is this happening? Clearly, all of these cities are different from one another. How can we be so certain that our hyperparameters for K-Means and/or DBSCAN can not be tweaked sufficiently to unveil a hidden cluster arrangement? Well, tSNE Clustering can help us see the inherent structure of the datapoints as a collective, despite them being in a 278-dimensional vectorspace; we will visualize in R2 where all of our points are in relation to one another before they’ve had a chance to be *designated* with a label by K-Means/DBSCAN. The results are as follows (with each point labeled as the actual city it belongs to, not a label designated by tSNE):

tSNE Plot of all Neighborhoods Across All Cities

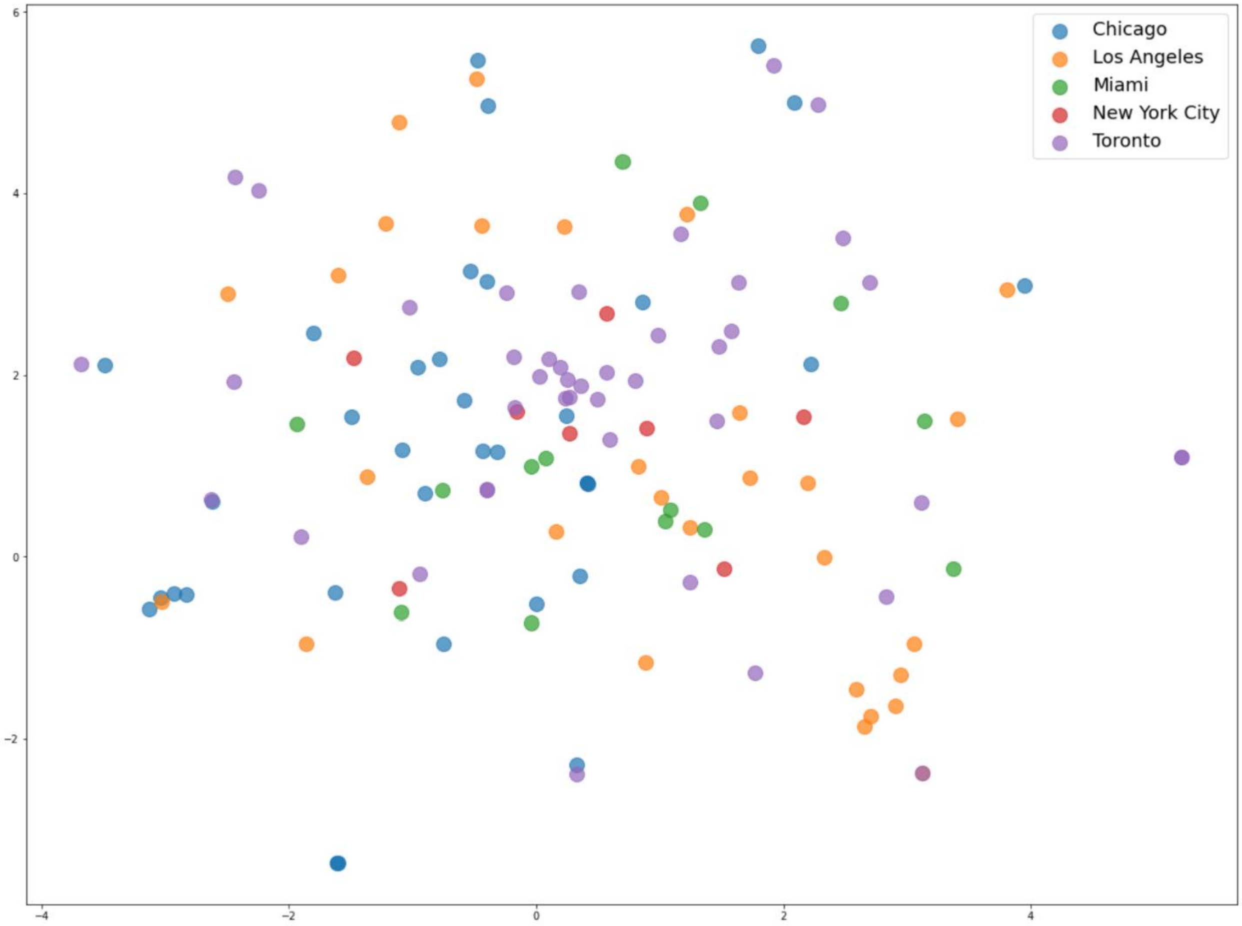
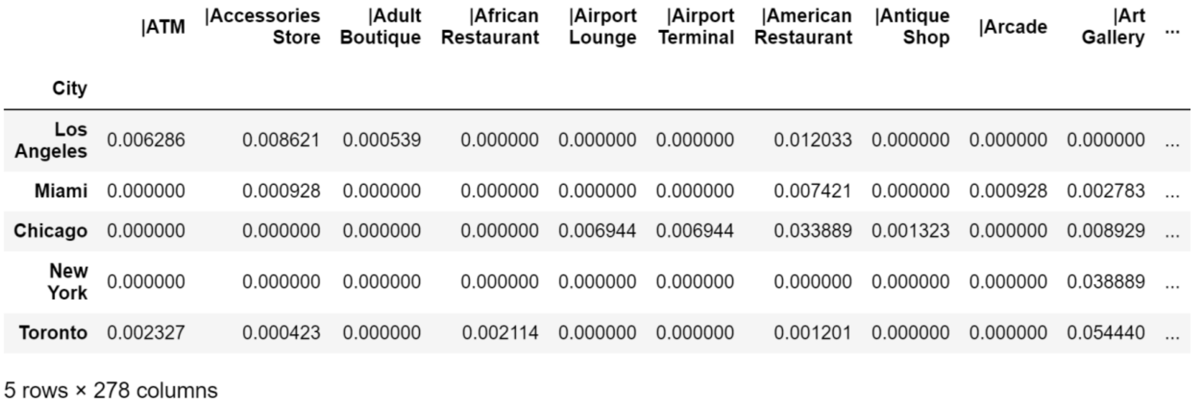


Figure 6. A t-Distributed Stochastic Neighbor Embedding (tSNE) plot of the neighborhood datapoints in all 5 cities. Each datapoint consists of 278 features (i.e., venue category frequencies), but this plot reduces that hyperdimensionality down to R2 for us to visualize based on points’ Euclidean distances from one another.

As can be seen, there’s no set pattern to where each city’s neighborhoods begin and where another city’s neighborhoods begin. There is a lot of overlap happening throughout, aside from a handful of points here and there that are from the same city. Had we seen this upfront, it may have been apparent that neither K-Means nor DBSCAN would be sufficient to encompass any one city’s characteristics reliably. This tSNE plot supports the fact that both Clustering algorithms lumped most neighborhoods into a single cluster or not at all, because there are no distinct clusters, essentially.

But finally, if we can not reliably *predict* which city is most like Toronto via Clustering, then *just how quantitatively similar* ***is*** *each city to Toronto?* It’s perhaps reasonable that different neighborhoods would have much the same venue fingerprint, technically, but most of the neighborhoods are not *exactly* the same. This is where Cosine Similarity comes into the picture. Recall the dataframe in Figure 2 in which each neighborhood could be described by the frequency with which each venue category was within its vicinity. Let's take that same dataframe and group it by city, with cell values being the result of the mean frequency of its corresponding venue category.



City	|ATM	|Accessories Store	|Adult Boutique	|African Restaurant	|Airport Lounge	|Airport Terminal	|American Restaurant	|Antique Shop	|Arcade	|Art Gallery	...
Los Angeles	0.006286	0.008621	0.000539	0.000000	0.000000	0.000000	0.012033	0.000000	0.000000	0.000000	...
Miami	0.000000	0.000928	0.000000	0.000000	0.000000	0.000000	0.007421	0.000000	0.000928	0.002783	...
Chicago	0.000000	0.000000	0.000000	0.000000	0.006944	0.006944	0.033889	0.001323	0.000000	0.008929	...
New York	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.038889	...
Toronto	0.002327	0.000423	0.000000	0.002114	0.000000	0.000000	0.001201	0.000000	0.000000	0.054440	...

5 rows × 278 columns

Figure 8. The venue category frequency makeup of each of the 5 cities, including Toronto.

We want to know which of the 4 other cities is most like Toronto, and we now have individual vectors of each city to mathematically compare. A single metric to accomplish this, instead of an algorithm, is to compute the Cosine Similary which is simply the dot product between two vectors divided by the product of their magnitudes. Doing so reveals the following:

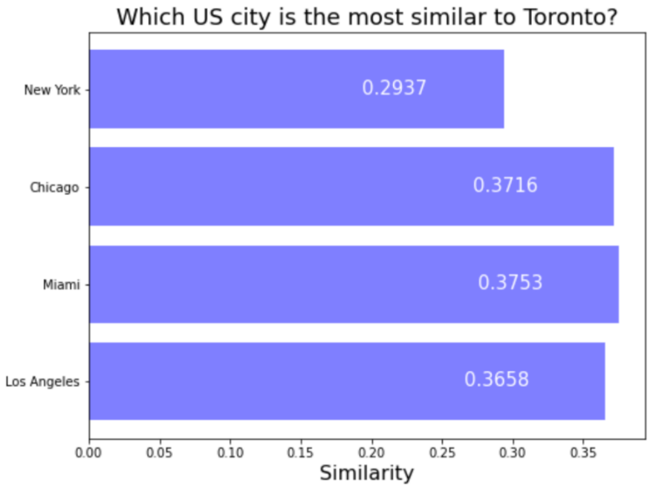


Figure 9. The Similarity scores for each city's venue categories when mathematically compared to Toronto.

What can be gathered from this is that Chicago, Miami, and Los Angeles are almost equally similar to Toronto, and that New York's similarity isn't that far off. It's worth noting that New York had the fewest number of neighborhoods in the dataset, hence a more complete list may prove itself to be much closer to the others, given the same set of features. Another thing to notice is that all of them have a modest, though substantial amount of Similarity to Toronto. While they may not be Similar in ALL the same ways (i.e., unique-yet-overlapping vector subspaces), it is worth recognizing that most neighborhoods have much the same types of places nearby usually.

DISCUSSION. So where should John Doe move to? Well, as it stands, the data does not reveal enough of a pattern to truly say. Yes, we knew from the beginning that the true spirit of a city can not be defined by venues alone, but it is nonetheless interesting to begin putting some structure to its answer. The short answer is that this study demands a part two that 1) consists of far more neighborhoods than venue categories, since having such a wide dataframe may explain why the tSNE plot essentially looks like noise; 2) incorporates information beyond venue categories, such as the ratings of the corresponding venues, how long each of them has been open, population variability, etc.

It is reasonable that merely a few neighborhoods sampled at random would be much the same, given merely venue category information. A more feature-rich and/or more geographically inclusive dataset would likely reveal a much more class-imbalanced map. The algorithms presented here only ended up having 122 neighborhoods available across all 5 cities (due to Nominatim/Foursquare limitations), which is miniscule given the 250+ features. Having fewer observations than features may be the sole culprit of the data not having much clusterable structure to it. But assuming the more single-clustered results are fairly accurate, given the venue data, it makes sense that all cities would have similar Similarity. For instance, it is very typical that any given part of town has their share of restaurants, banks, gyms, bars, etc. While the specific makeup of venue types is technically unique for each neighborhood, the majority of any given city’s vectorspace is likely to overlap with a lot of the same types of places that every neighborhood needs (i.e., things like restaurants, grocery stores, convenient stores, and gas stations can be found everywhere).

CONCLUSION. The data provided is insufficient to resoundingly recommend that John Doe move to one city over the others. The K-Means and DBSCAN Clustering both quite definitively offer no insight into the differences between the cities. Cosine Similarity offers the most insight, with it at least allowing us to declare New York as the least like Toronto; however, the dataset here showed New York to have the fewest number of neighborhoods, which perhaps skews the results. That said, if the other cities are any indication, then more New York neighborhoods in the dataset may reveal it to be just as Similar as the other cities (in terms of venues categories, alone). Further study is recommended with 100 or more neighborhoods from each and every city, more selective reduction in venue categories available, and additional features related to venue, safety, education, and his job. While John Doe was only interested in lifestyle comparison, more metrics are needed to provide him with a robust, and well-rounded recommendation on where to move.

Corresponding JUPYTER Notebook containing Python code may be found here:

https://github.com/christianharrisDS/Coursera_Capstone/blob/master/SimmedCities%20IBM%20CAPSTONE.ipynb