# **Determining the best neighbourhood to expand business operations**

# Raynell Foo

# July 21, 2020

## Introduction

* 1. The Problem

A business selling kitchen ingredients to food places in New York is thinking of expanding its operations to another city. It wants to open its operations in Toronto but is unsure in which neighbourhood it should house its office which also doubles as a warehouse. The ideal location would be in a neighbourhood surrounded by the most food places, that way it can maximise visibility among its potential clients as well as reduce transportation distance between the food places and the business' office/warehouse, which will ultimately reduce delivery cost.

* 1. The Task

To help determine the ideal location for its new office/warehouse, I have been tasked to utilise Foursquare API to determine the most suitable neighbourhood for the business to open in.

## Data Acquisition and Cleaning

2.1 Data Sources

Location data in Toronto which includes a table of postal codes and boroughs for all neighbourhoods in Toronto are sourced from a Wikipedia page [here](https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M). Longitude and latitude data for all the neighbourhoods are sourced in an online downloadable [here](http://cocl.us/Geospatial_data). All venue locations and their details are accessed from Foursquare’s API, which can be called from [here](https://api.foursquare.com/v2/venues/explore?), and their venue category hierarchy is manually scraped from Foursquare’s website [here](https://developer.foursquare.com/docs/build-with-foursquare/categories/).

2.2 Data Cleaning

The location data downloaded from Wikipedia are transformed into a Pandas DataFrame. The DataFrame is then cleaned by removing all rows with unassigned borough, and later renaming the postal code column to ease a later DataFrame combination, followed by resetting the index. All neighbourhoods with an unassigned value are replaced with the name of its borough.

The longitude and latitude data for all neighbourhoods are downloaded and read into a Pandas DataFrame. It is then merged with the previous DataFrame generated from Wikipedia using a Left Join merge.

A DataFrame of all venues within a 2-kilometer radius from all neighbourhoods is created using a function that pulls the data using Foursquare’s API. The DataFrame consists of all venues from all categories, and we are only interested in venues that fall under the food umbrella category. To filter the DataFrame so that it is relevant to us, a list of venue categories that falls under the food umbrella is scraped from Foursquare’s website. The raw html scraped is cleaned through creating a Beautiful soup object that is parsed using an html parser, then filtered using a nested for loop that extracts the name of the relevant venue categories from its <h3> html element. The cleaned Beautiful soup object is then appended to an empty list and indexed so that it contains only all venue categories that falls under the food umbrella.

The DataFrame of all venues is then filtered using the cleaned list into a new DataFrame, and its index is then properly reset. The DataFrame is then grouped by its neighbourhood, which shows the counts of all food venues in each neighbourhood. This grouped DataFrame is then sorted in descending order, so that the neighbourhood with the most food venues is arranged to be at the top.

## Python Libraries Utilised

3.1 Pandas and Numpy

All data exploratory and cleaning are done with these 2 Python packages. The main data organisation work is conducted using Pandas DataFrame. Pandas is also used to read the table from the Wikipedia page into a DataFrame using the “read\_html” function, and a CSV file into a DataFrame using the “read\_csv” function, which saves time and lines from having to scrape the data manually.

3.2 Lxml and Requests

These 2 libraries are used to assist the Pandas library to parse data into a DataFrame as well as create API calls to Foursquare.

3.3 Beautifulsoup4

This library is used to scrape the venue category hierarchy data manually from Foursquare’s website, as an API call is unable to be made due to lack of premium credentials. The raw html data is parsed into a Beautiful soup object and filtered further using nested for loops and the “.find” and “.find\_all” functions.

## Conclusion

Through the usage of data visualisation and exploratory analysis using the various Python libraries, we are able to find the neighbourhood with the highest amount of food venues in Toronto. The business should open its operations in that neighbourhood to maximise the benefits as discussed in the above sections.