

IBM Applied Data Science Capstone

Tourism Finder

A. Alkaff Ahamed

14th June 2020

Contents

1. Introduction
2. Data
3. Methodology
4. Results
5. Discussion
6. Conclusion

Introduction

Define Tourism

Use Cases (Audience)

Scope of Project

Define Tourism

- Tourist Attraction:
 - A place of interest where tourists visit
- Tourism Hotspot:
 - A city with a group of tourism attractions
- Significance:



Use Cases (Audience)

- Almost any business
 1. Fancy Restaurants
 2. Nightclubs
 3. Hotels and Star Hotels
 4. Malls
 5. Souvenir Shops
 6. And more...
- Planning vacations

Scope of Project

- User will input city name
- Application will output a map
 - Map is interactive
 - Automatically zoomed to the region
 - Tourist attractions are marked on the map

Data

Libraries and APIs used as Data Source

Libraries and APIs

1. GeoPy Library

To obtain coordinates of city

2. Foursquare API

To obtain tourist attractions

3. Folium Library

To plot the points on the interactive map

Methodology

Retrieve Coordinates of the City

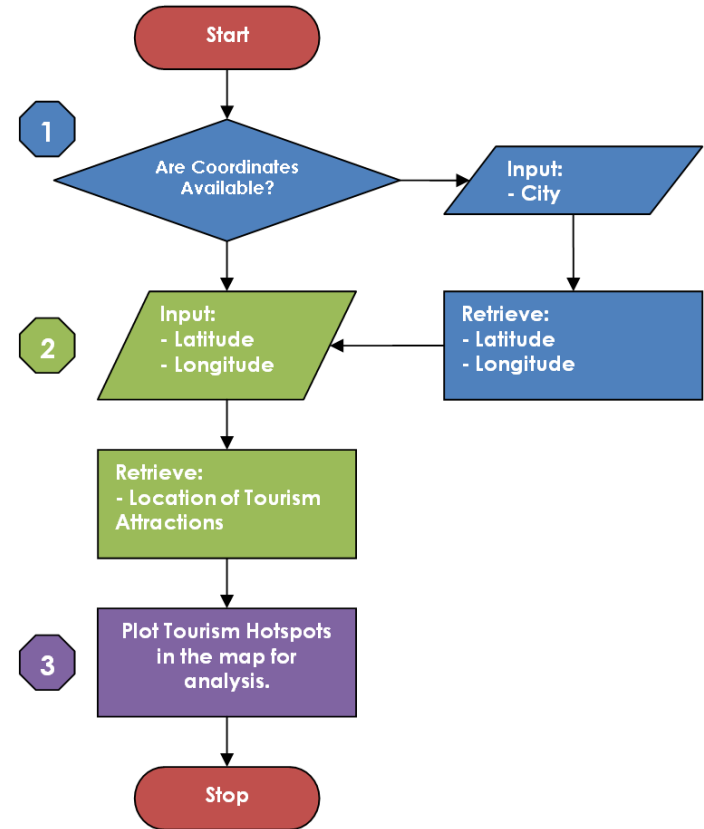
Retrieve Location Data of Tourist Attractions in the City

Plot the Tourism Attractions in the Map

Methodology

3 Main Steps

1. Retrieve coordinates of the city
2. Retrieve location data of tourist attractions in the city
3. Plot the tourism attractions in the map



Retrieve Coordinates of the City

- This step can be skipped if user has the coordinates
- If user doesn't have coordinates,
 1. Input the city name
 2. Run the code
 3. Obtain coordinates and continue

Retrieve Coordinates of the City

1. Import the library
2. Create a geolocator object
3. Create a location object
4. The attributes latitude and longitude of location object will contain the coordinates we need

Retrieve Location Data of Tourist Attractions in the City

1. The coordinates will be fed into Foursquare API
2. Search query is run to obtain tourism attractions
 - Result will be JSON file
3. Convert to data frame
4. Clean up the data

Plot the Tourism Attractions in the Map

1. Use folium and initialise the map with same coordinates
2. Run a for loop through all the tourism attractions
3. Add marker in each iteration
4. Display the map and explore

Results

Retrieve Coordinates

Retrieve Location Data of Tourist Attractions

Plot the Tourism Attractions

Retrieve Coordinates (Own Coordinates)

Dashboard

You will set parameters here. In fact, this dashboard can be replaced with any other kind to get the user input, example, Command-Line Interface, GUI.

```
# Set city name
city = "Singapore"

# Or set Latitude and Longitude manually
# and set city to None by uncommenting the line
lat = 13.0827
lon = 80.2707
city = None
```

Step 1: Retrieve coordinates

```
if city != None:
    geolocator = Nominatim(user_agent="foursquare_agent")
    location = geolocator.geocode(city)
    #print(type(location))
    lat = location.latitude
    lon = location.longitude
    print ("The coordinates of %s are (%f, %f)." % (city, lat, lon))
else:
    print ("Your chosen coordinates are (%f, %f)." % (lat, lon))
```

Your chosen coordinates are (13.082700, 80.270700).

Retrieve Coordinates (Using City)

Retrieve Coordinates

- The difference will be in the output
 - By city: The coordinates of [City] are...
 - By user coordinates: Your chosen coordinates are...

Retrieve Location Data (Downloading)

Step 2: Retrieve list of tourism attractions

Retrieve by using Foursquare API Call

```
search_query = 'Tourism'
radius = 50000
url = 'https://api.foursquare.com/v2/venues/search?client_id={}&client_secret={}&ll={},{}&v={}&query={}&radius={}&limit={}'.format(
    search_query, radius, search_query, search_query, search_query, search_query, search_query, search_query)
results = requests.get(url).json()
print("Status code: %d." % results['meta']['code'])
```

Status code: 200.

Retrieve Location Data (Converting to Dataframe)

Process the JSON file

```
# Assign relevant part of JSON to venues
venues = results['response']['venues']

# Transform venues into a dataframe
dataframe = json_normalize(venues)
print("%d Results loaded." % dataframe.shape[0])
print("Displaying first 3 rows:")
dataframe.head(3)
```

50 Results loaded.
Displaying first 3 rows:

	categories	hasPerk	id	location.address	location.cc	location.city	location.country	location.crossStreet	location.di
0	[[{'id': '4bf58dd8d48988d12c951735', 'name': 'E...'}]]	False	4cff4c65f7b38cfa9a7cc8c3	370 Orchard Road	SG	Singapore	Singapore	NaN	
1	[[{'id': '52e81612bcbc57f1066b7a13', 'name': 'N...'}]]	False	52e0d6f4498e23eedfbea071	Mandai Lake Road	SG	Singapore	Singapore	NaN	
2	[[{'id': '4bf58dd8d48988d1f6931735', 'name': 'G...'}]]	False	4bfa092f8f32ef3bd0cf04aa	80 Robinson Rd	SG	Singapore	Singapore	NaN	

Retrieve Location Data (Cleaning)

Fix the Categories column

```
# Fix the Categories Column

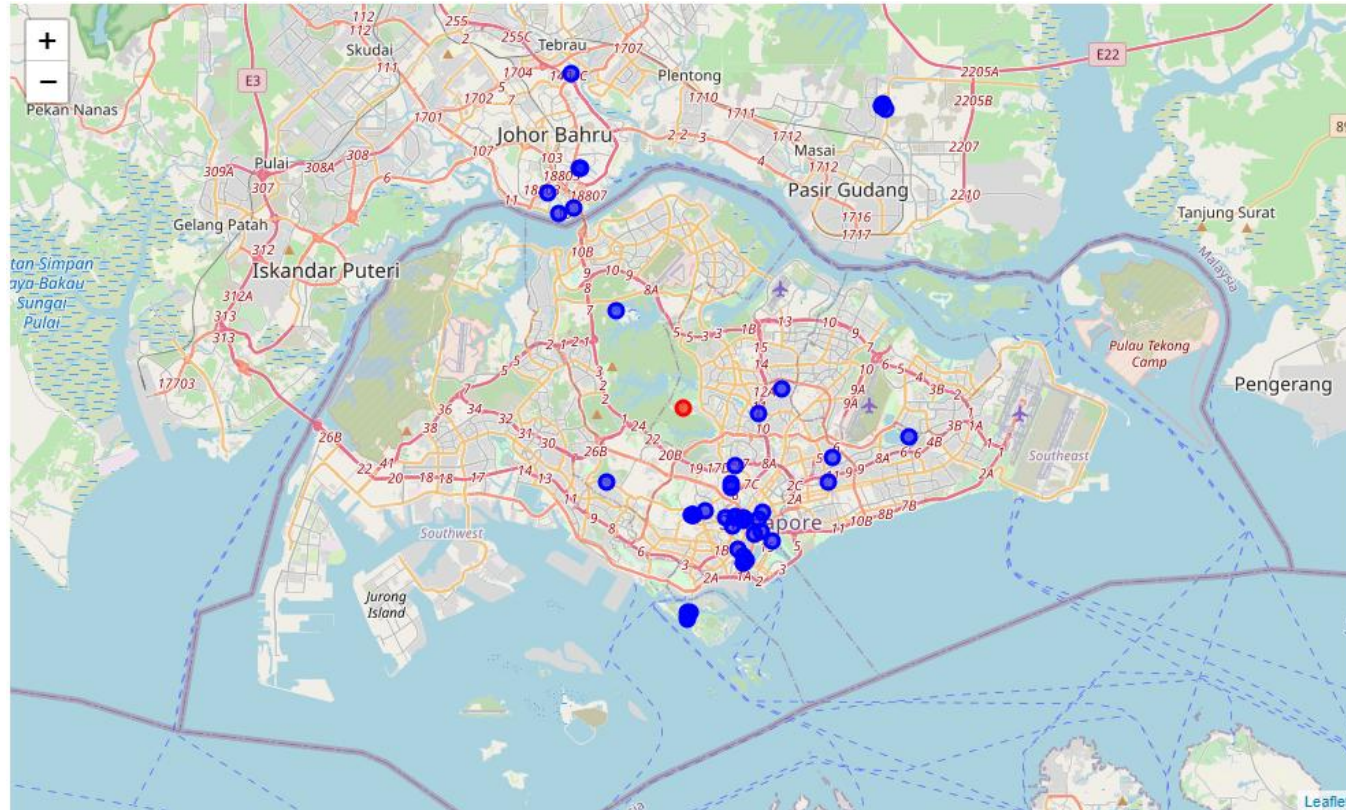
# Function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']

# Apply filter to the categories column
dataframe['category_type'] = dataframe.apply(get_category_type, axis=1)
dataframe.head(3)
```

	categories	hasPerk	id	location.address	location.cc	location.city	location.country	location.crossStreet	location.distance	location.fv
0	Embassy / Consulate	False	4cff4c65f7b38cfa9a7cc8c3	370 Orchard Road	SG	Singapore	Singapore	NaN	5817	[370 Orche
1	Nature Preserve	False	52e0d6f4498e23eedfbae071	Mandai Lake Road	SG	Singapore	Singapore	NaN	6621	[M
2	General	False	4bfe002f0f22e23e2bd0e0f04a3	90 Robinson Rd	SG	Singapore	Singapore	NaN	0220	

Plot the Tourism Attractions



Discussion

Discussion (Problem)

- Foursquare API returned only some tourism attractions correctly
- Additionally, some tourism agencies were returned
- These cause inaccuracy in application function

Discussion (Solution)

- The query string can be changed
- Multiple queries can be run and results be combined
- Machine learning models can be used to detect these errors

Conclusion

Conclusion

- Overall, the application identified tourism locations correctly
- Only the attractions in the locations were not identified correctly
- This inaccuracy problem can be fixed in next version
 - By using machine learning
 - By running multiple queries

THE END

Done by A.Alkaff Ahamed