EE810H

Restaurant Data Map Project Report

Chang Liu(10373152), Zeyu Wang(10376853), Chengzhao Li(10374848)
Instructor:Professor Mukund Iyengar
May 4, 2014

ABSTRACT

This project collected different restaurant informations around New York City,including restaurant name, rating value, address, then demonstrated them in forms of different data maps. Through the visualized data, a more intuitive and completed result can be generated from one single search compared to result from ordinary website. Provided a macroeconomic view of information such as the locations, rating values and distribution density of restaurant in New York area. In addition, a graphic user interface was implemented for better interactive experience.

1 INTRODUCTION

We are living in an interconnected world. Large amount of information could be easily access from the Internet. For example, you can get as many as 47947 results when browsing Yelp.com for restaurants in New York City area. Traditional applications bring people lots of convenience when finding the restaurant they want. But when a user wants to make a choice with comprehensive information, traditional applications' functionality became limited. For instance, when a user wants to find an area that has three specify kinds of restaurant, he has to search three times and compare the data manually. Or, if a traveler wants to find a place with the highest density of one cuisine, he has to zoom in and zoom out several times and do the comparison by himself. In our research, we develop software that plotting different kinds of restaurants from one search result in different colors on a geography map. Users could distinguish restaurant type easily from the color and find the location that has all restaurants

he need without any other operation. Also, users could know the density of a certain kind of restaurant from our application.

Our system is separated into three modules: data collecting, data visualizing and graphic user control panel. Data collecting module is response for grabbing restaurant information from the Internet and save the information into several files. Data visualizing module utilizes the data collected from the Internet, process it and create the data map. Graphic user control panel creates a graphic panel for users to control what kinds of data will be plotted on the map.

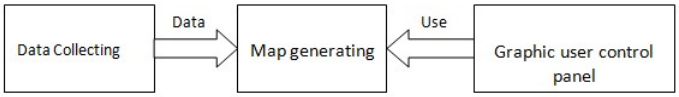

Fig 1.1 System Architecture

2 Data Collection

Restaurant information are needed from Yelp, including name, ratings, number of reviews, price range and addresses.

2.1 Web Scraping

- URL Analysis
  - A Uniform Resource Locator(URL), as a specific character string,constitutes a reference to a resource.
  - The source from Yelp has a common format. According to its URL, restaurant information are listed as a regular pattern.
  - Here is an example of Indian food in Jersey City that we searched from Yelp, and some of the data from its URL are listed below.

```
<div class="search-result natural-search-result biz-listing-large" data-key="1">

<img alt="Sapthagiri Taste Of India" class="photo-box-img" height="90"
src="//s3-media2.ak.yelpcdn.com/bphoto/dhKmqmBdEmc4OvCceEeb-g/90s.jpg"
width="90">
<span class="indexed-biz-name">1. <a class="biz-name">
href="/biz/sapthagiri-taste-of-india-jersey-city#query:indian%20restaurant"
data-hovercard-id="gQw3yAXJMI3ApbtecoeiHg">Sapthagiri Taste Of India</a>
</span>
<i class="star-img stars_4" title="4.0 star rating">
<img alt="4.0 star rating" class="offscreen" height="303"
src="http://s3-media3.ak.yelpcdn.com/assets/2/www/img/c2252a4cd43e/ico/stars/v2/s
tars_map.png" width="84">
<span class="review-count rating-qualifier">
148 reviews
</span>
<span class="business-attribute price-range">$</span>
<div class="secondary-attributes">
<address>
804 Newark Ave<br>Jersey City, NJ 07306
</address>
key = str(MediaBox.get('data-key'))
```

Fig 2.1 URL of Indian food in JC from Yelp

  - From the data we receive throgh the URL from Yelp, the information we need are high lighted above.

- Browser Simulation
  - In this paper, we use Python urllib2 module to open the URL of Yelp.
- Data extraction
  - A Python library called Beautiful Soup are adopted for scraping. Some of the code are listed below, so that we can extract the information that needed.

```
Name = str(MediaBox.find(attrs = {'class':"photo-box-img"}).get('alt').encode('ascii',
'ignore'))
AddressList = str(MediaBox.find(attrs =
{'class':"secondary-attributes"}).address).replace('<br/>','\n').split('\n')
Rating = str(MediaBox.find('i').get('title'))

ReviewCount = str(MediaBox.find(attrs={"class":"review-count
rating-qualifier"}).get_text().strip())
try:
    PriceRange = str(MediaBox.find(attrs={'class':"business-attribute
price-range"}).get_text().strip())
```

Fig 2.2 Beautiful Soup codes for collecting data

2.2 Address Recognition

- As we have collected all the restaurant addresses, it is necessary to locate all the addresses in the map.
- We convert all the addresses into geographic coordinates (latitude and longitude) using Google Geocoding API.

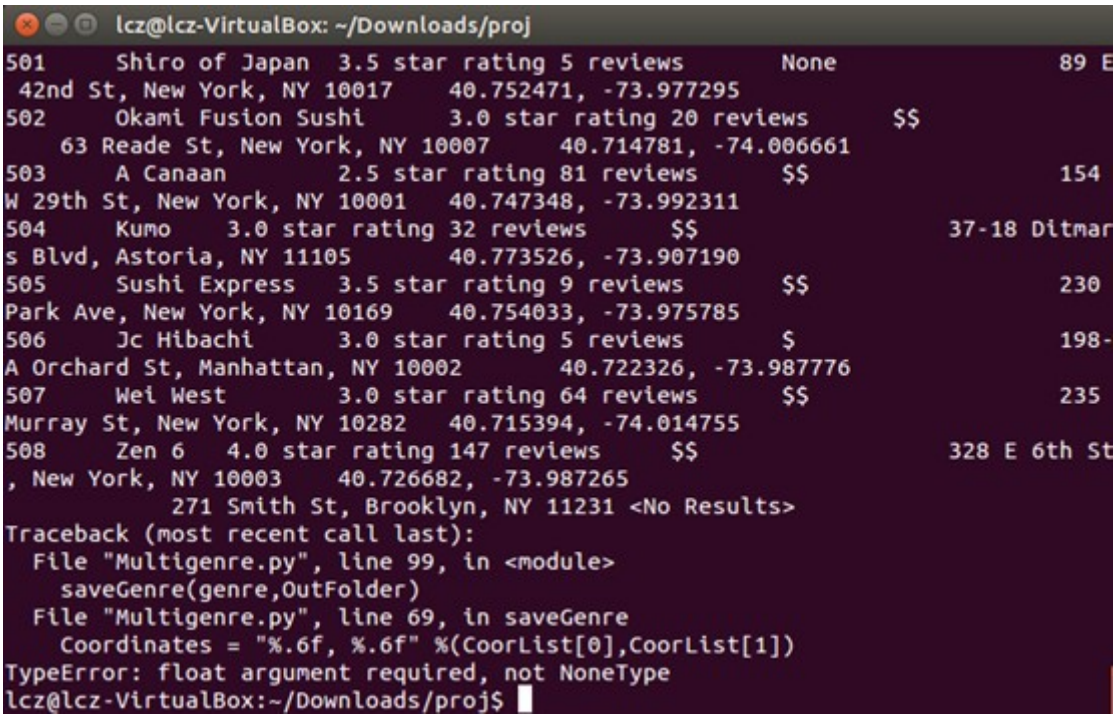

Fig 2.3 Screen shot when running codes

- However, in order to prevent abuse of the service, a user can submit only 2500 requests per 24 hour period from Google Geocoding API. As the screen shot above showed, no results is returned for the last restaurant address, which means requests have reached their limit.

- Finally, we save different cuisine of restaurant information extracted in different textfile documents.

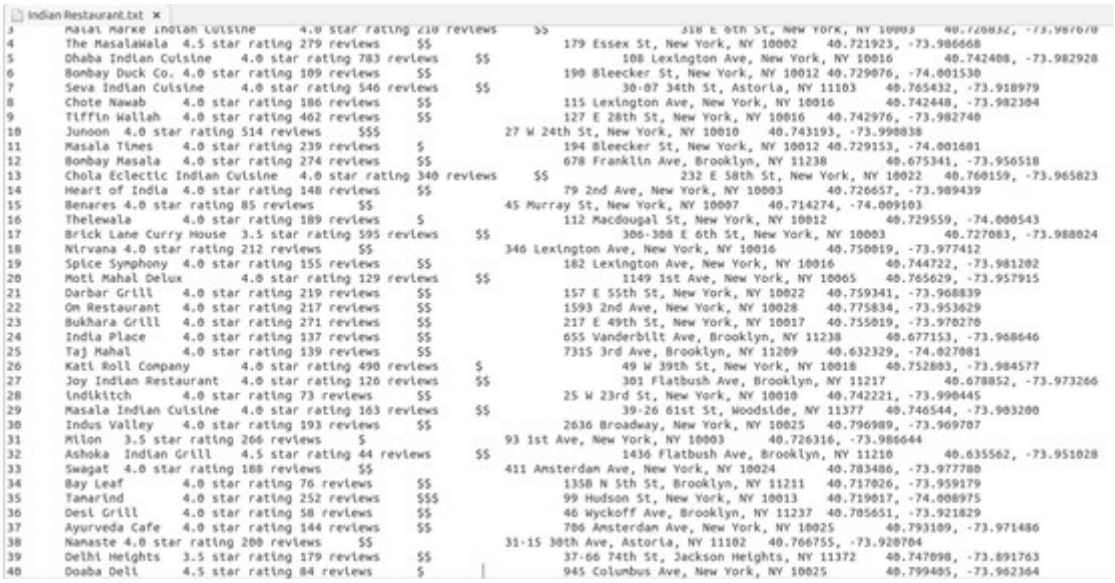

Fig 2.4 saved data in textfile

3 Data Visualization

Four kinds of maps were implemented to visualize all the gathered data: The first one shows the location of restaurants drawing on the basic map(Location Map); The second one shows the density distribution of restaurants drawing on the basic map(Density Map) ; The third one shows the rating of restaurants on their locations drawing on the basic map(Rating Map A) and the last one shows the rating of restaurants on their locations drawing on the google map background(Rating Map B).
The examples of each kind of maps are shown in fig.1 to fig.4.

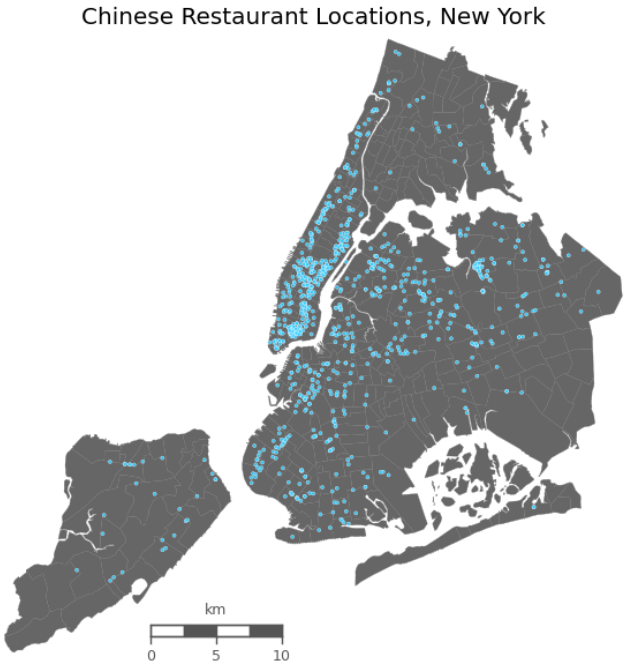

Fig.1 location map

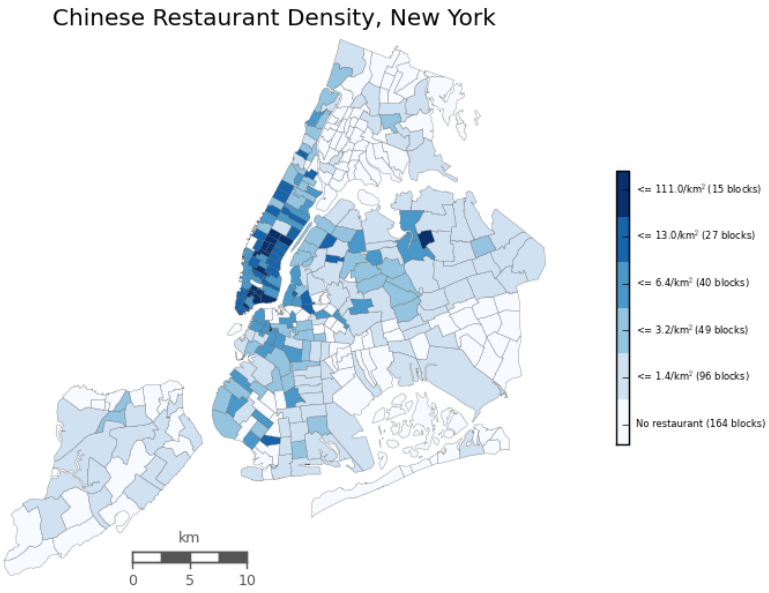

Fig.2 density map

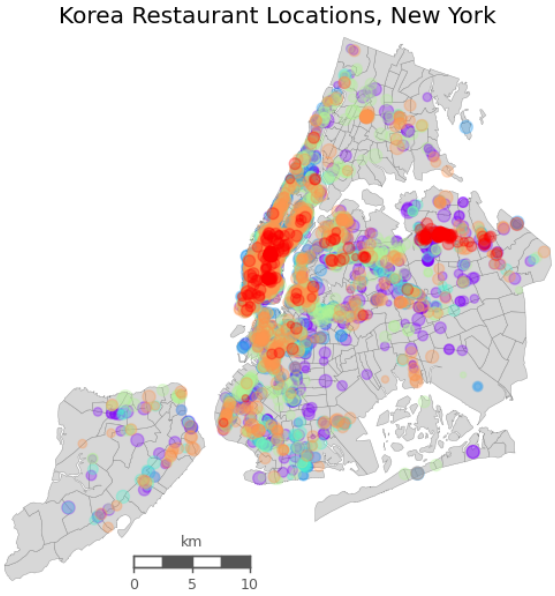

Fig.3 rating map A

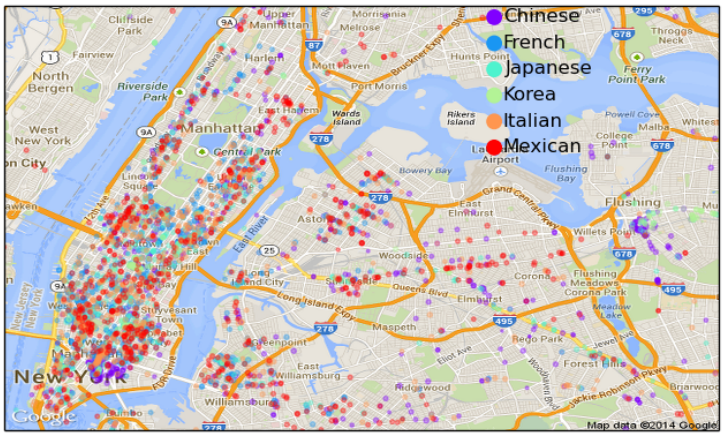

Fig.4 rating map B

Note that, in fig.1, one point represents one restaurant; In fig.2 and fig.3, one point represent one restaurant, the area of one point represents the rating value of the correlated restaurant,

different colors represents different kinds of restaurants; In fig.4, basic map are divided into many blocks, the color of one block represents the density of restaurants of the block, to be specific, blocks of the darkest color has the highest density of restaurants, blocks of the lightest color represents the lowest density of restaurants.

3.1 ENVIRONMENT REQUIREMENTS

- **Package**
  Four independent modules were coded for plotting four kinds of map, respectively. For running each of these modules, the following non-stdlib are required :
  - **Pandas**:Provides high-performance, easy-to-use data structures and data analysis tools for python
  - **Numpy**: The fundamental package for scientific computing with python
  - **Matplotlib**: A python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms.
  - **Basemap**: A Matplotlib toolkit, library for plotting 2D data on maps in Python
  - **Shapely**: A package for manipulation and analysis of planar geometric objects.
  - **Fiona**: Provides uncomplicated Python interfaces to functions in OGR
  - **Descartes**: Use geometric objects as matplotlib paths and patches
  - **PySAL**: a spatial analysis library
- **Shapefiles**
  A New York City shapefile is also required for plotting the graph. The shapefile contains data such as map boundaries and polygon datas that divide the map into different blocks.
- **Restaurants' rating, longitude and latitude**
  Rating Values and Coordinates of restaurants are required for plotting points.

3.2 VISUALIZATION PROCESS OUTLINE

The Ouelines for plotting each map.

3.2.1 PLOTTING A LOCATION MAP

1. Extract boundary info using fiona from New York City shapefile:
2. Create a Basemap instance which works like a canvas that we can draw the map on,
3. Extract Polygon objects from the shapefile: each Polygon objects describes a block on the map.
4. Extract coordinates of each restaurant, Transform them into Point objects.

5. Filter each Point object out if it doesnâĂŹt fall into the New York City map.
6. Transform each Polygon objects to Patch object.
7. Draw Patch objects on the Basemap instance.
8. Draw Points objects on the Basemap instance

3.2.2 Plotting a Density Map

1. Extract boundary info using fiona from New York City shapefile:
2. Create a Basemap instance which works like a canvas that we can draw the map on,
3. Extract Polygon objects from the shapefile: each Polygon objects describes a block on the map.
4. Extract coordinates of each restaurant, Transform them into Point objects.
5. Count Point objects that fall into each Polygon object.
6. Normalize count values for each Polygong object.
7. Transform each Polygon objects to Patch object, correlated count value determines the color of the Patch object.
8. Draw Patch objects on the Basemap instance.

3.2.3 Plotting a Rating Map A

1. Extract boundary info using fiona from New York City shapefile:
2. Create a Basemap instance which works like a canvas that we can draw the map on,
3. Extract Polygon objects from the shapefile: each Polygon objects describes a block on the map.
4. For each genre of restaurant
   a) Extract Both coordinates and rating value of each restaurant, Transform them into Point objects.
   b) Filter each Point object out if it doesnâĂŹt fall into the New York City map.
   c) Draw Points objects on the Basemap instance. for each Point object, genre determines the color of the point, rating value determines the diameter of the point.
5. Transform each Polygon objects to Patch object.
6. Draw Patch objects on the Basemap instance .

3.2.4 PLOTTING A RATING MAP B

1. Request a map image in png format from google.com
2. Create a Basemap instance which works like a canvas that we can draw the map on, For each genre of food
3. Extract Both coordinates and rating value of each restaurant, Transform them into Point objects.
4. Filter each Point object out if it doesnâĂŹt fall into the New York City map.
5. Draw Points objects on the Basemap instance. for each Point object, genre determines the color of the point, rating value determines the diameter of the point.
6. Show the png image as background on the Basemap instance.

4 GRAPHIC USER PANEL

After finish plotting restaurants information on geography map, the next step for us is designing a graphic user control panel.

4.1 GUI LIBRARY SELECTION

There are certain kinds of Python GUI libraries in the market includeïijŽ

- PyQt
- Pyside
- PyGTK
- wxPython
- TkInter

Pyside became our choice for its functionality and maintainability. PySide is an open source software project providing Python bindings for the Qt framework. Qt is a cross-platform application and UI framework, allowing the developers to write applications once and deploy them across many operating systems without rewriting the source code, while Python is a modern, dynamic programming language with a vivid developer community. Combining the power of Qt and Python, PySide provides the wealth of Qt framework for developers writing software in Python and presents a first-class rapid application development platform available on all major operating systems(http://qt-project.org/wiki/About-PySide).

4.2 User Panel Design

We include four buttons and six checkboxes in our user panel which is shown is Fig. 4.1. The buttons controls the type of map to be generated; the checkboxes indicates restaurantsâĂŹ nationality.

Fig 4.1

1. 'Show Basic' and 'show Google' allows users to select multiply restaurant type at the same time. To be specific, checking 'Show Google' button with the selection of Chinese and Italian would bring the user a restaurant information map with all Chinese restaurants and Italian restaurants in Google style (See Fig. 4.2).

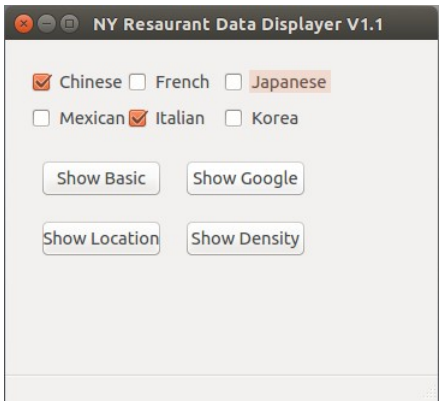

Fig 4.2

2. 'Show Location' and 'Show Density' requires users select only one restaurant type each time. If a user selects more than one type for these two functions, he will get an warning of 'Only Accept One Input' at the bottom of the panel. For instance, if I select both Korea and French and check 'Show Density Map', no map would be display and a warning message would be shown (See Fig. 3).

Fig 4.3

5 EVALUATION

(1)In Data Collection part, we gathered all the data by extracting from Yelp.com. Because the usage limitation of Google Geocoding API, we can only gather no more than 2500 useful data per day, which limited our research region to only New York City area and only 6 genres of restaurant. If there is a better data source we can apply the research result on wider area.
(2)In Data Visualizition part, if we can implement a interactive data map, the result can be more useful.

6 CONCLUSION

In our research, we offer a static solution for showing all restaurant information of a certain area in one signle map. We use urllib2 and Beautiful Soup to collect data from Yelp website; use basemap toolkit to generate the map together with visualized restaurant data; use PySide lib to create user control panel. For the next step, we want to (1)expand the scale to nationwide, (2)use database instead of text files to organize data and (3)implement an interavtive data map with more information showed for the restaurant.