DataW

Owing to make our visualization having great society value, we choose the Chinese housing price topic which is the hot spot and difficult problem in nowadays Chinese biggest city such as Beijing and Shanghai. Connecting to our CS graduate student vital interests, we ask a leading question: When a CS graduate student can buy a 70 square meter home in Beijing? And our project is focused on giving a directly answer to that question. We believe that every Chinese CS graduate student will face and interest in this question.

To achieve our objective, three main process are listed as follow:

I. Data Wrangling

1. Capture and parser House price data in Lianjia website

Input: Lianjia Web: http://bj.lianjia.com/xiaoqu/

Script: LianjiaSpider_0_1.py (District Capturer)

LianjiaSpider_0_2.py (Neighborhood Capturer)

dbbase.py (SQLite database entity)

Output : aptpricedb.db (SQLite database)

Software: Python 2.7 lib: urllib2, BeautifulSoup

OpenVpn (Change ip for LianjiaSpider_0_2.py)

Or

Abu Cloud: https://www.abuyun.com/

Detail:

LianjiaSpider_0_1.py

Acquire the District and Mini-District Url to Capture the neighborhood information for each area.

Run the LianjiaSpider_0_1.py to capture data.

LianjiaSpider_0_2.py

Acquire the neighborhood information for each area.

Lianjia Web had applied a IP blocker for 30mb stream data each hour.

My solution is use the Abu Cloud to use random ip request or use the openvpn to manually change ip.

The LianjiaSpider_0_2.py use the openvpn.

Run the LianjiaSpider_0_2.py in same folder with aptpricedb.db produced by LianjiaSpider_0_1.py. And change ip by using openvpn when the script shows information.

2. **Capture Beijing Salary Data from jobUi**

Input: JobUi Web: http://www.jobui.com/salary/beijing/

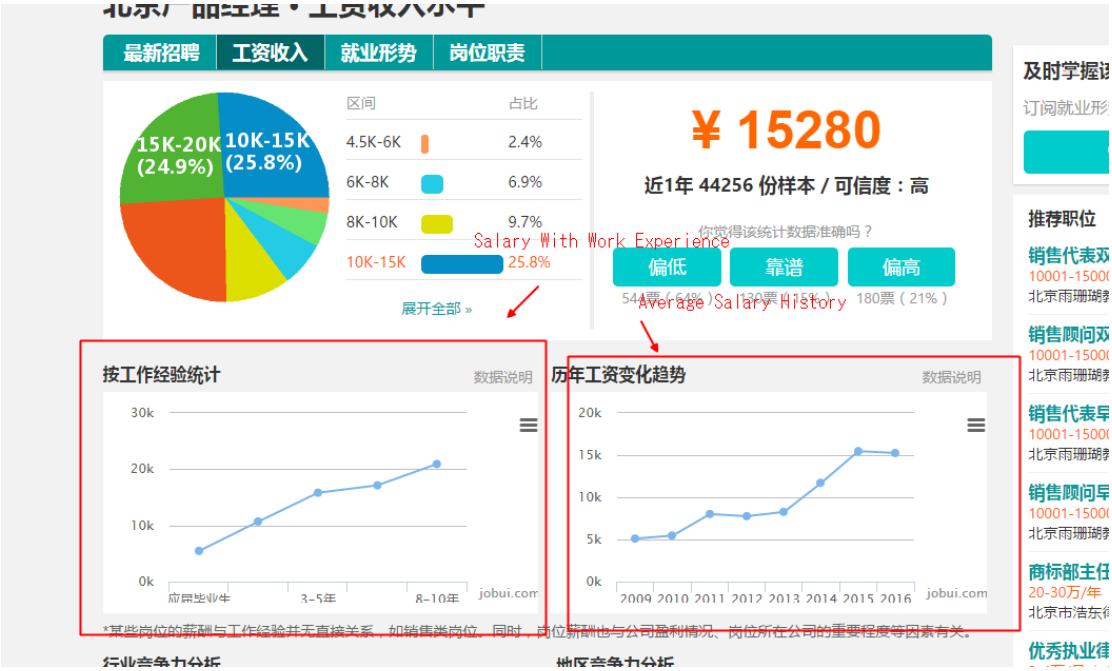
Script: JobUiSpider.py

Output : wage.csv

Software: Python 2.7 lib: urllib2, BeautifulSoup

Detail:

Capture the url for each job title.



Capture the salary history and Salary with different work experience.

Run the script JobUiSpider.py.

3. Capture Coordinate for each neighborhood

Input: google geo API

Baidu geo Api

aptpricedb.db(With neighborhood data)

Script: coodCap.py(Main script)

dbbase.py (SQLite database entity)

Output : aptpricedb.db(With coordinate for each neighborhood)

Software: Python 2.7 lib: geopy

Detail:

The script will get coordinate of each neighborhood for different combination of name, district and neighborhood information.

The script require 6 free Google API Key in different google project with geo API active or 2 paid Google API. And a baidu ApI Key. Key can be acquired by flowing address:

http://lbsyun.baidu.com/

https://console.developers.google.com/apis/

Place the Input and other Script in same folder

Input the both API key In script

Run the script.

II. Data Analysis

1. Analysis the Salary data and produce

Input: wage.scv(Captured Salary data from JobUI)

Script: WageAna.py

Output : samplewage.js(Salary factor in JSON to regenerate the Salary Curve)

Software: Python 2.7 lib: unicodecsv, numpy, matplotlib

Detail:

We select 3 main job catalog in computer science related job title.

By use the data from jobui we can regenerate the salary incensement curve.

After use the curve fitting the factor can be used for reproduce the prediction of total salary in certain years of work.

Place Script and Input in same folder and run script.

2. Analysis the neighborhood data and reproduce the heat map data

Input: Beijing.json (Beijing geometry data)

Aptprice.db(neighborhood data with coordinate)

Script: is_in_poly.py(main script)

dbbase.py (SQLite database entity)

Output : beijingdata.js (Apartment price and Beijing geometry data)

Software: Python 2.7 lib: unicodecsv, numpy, matplotlib

Detail:

Part the neighbor data with Beijing geometry data. Identify each neighborhood location in geometry data.

Capture and cleaning the house price for 70m^2 house from Aptprice.db.

Generate the average price data for each district.

Combating all data to a Json string that can read by the D3.

Put Input and Script in same folder and run the script.

III. Visualization

1. D3!

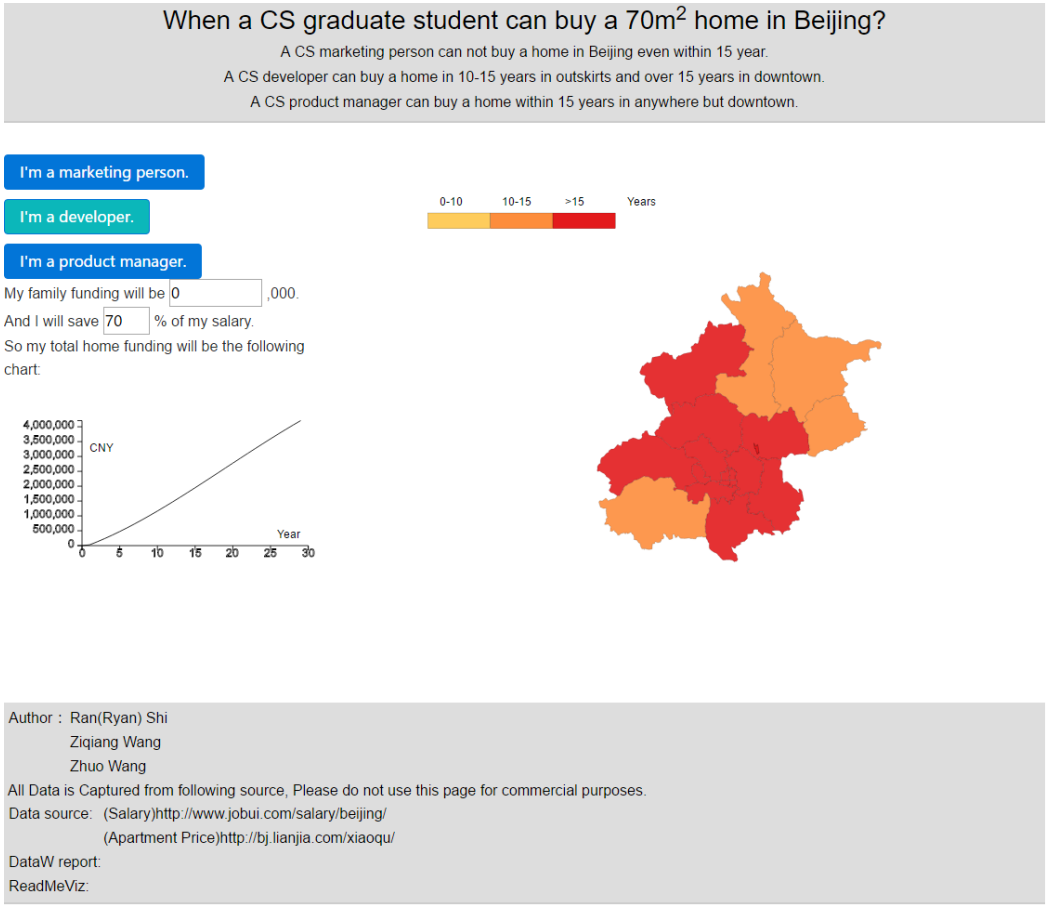
Input: beijingdata.js (Apartment price and Beijing geometry data)

samplewage.js(Salary factor in JSON to regenerate the Salary Curve)

Script: index.html

Output : index.html

Software: JS js: jquery,d3, tether bootstrap;css: bootstrap, tether



Implement the heatmap visualization with d3 by using geometry data and house

price form json data.

Built Cubic function to regenerate the total salary form the factor which is generate by curve fitting.

Build the button and input with dynamic rendering in d3 which allow user to select their or input on their own condition.

Implement the Salary visualization with d3.

Put index.html with /js and /css, then run index.html.