CS591 L1 Project Report

Yihong Guo

In this project, I make two HTML file: one shows the structure and size of different employment department in Boston and the other shows some information of people who have increasing earnings during the past three years.

I got the idea of this project when I was looking for possible datasets on City of Boston Data Portal. After I go through first several pages of datasets, I was attracted by the series of Employee Earnings Report. Then I come to an idea: if I combine and virtualize these datasets, is it possible to find something that is more easily for common people to understand the structure of employments in Boston and some potential information these datasets would show after virtualize them.

Datasets used:

Employee Earnings Report 2012

Employee Earnings Report 2013

Employee Earnings Report 2014

Datasets created:

Employment Structure 2014: a dataset that restructures “Employee Earnings Report 2014” and categorizes the information of employers by departments and titles. A sample of this would be:

Original structure:


```
"total_earnings" : "100381.19",
  "zip" : "02132",
  "detail" : "0.00",
  "injured" : "0.00",
  "title" : "Supvising Claims Agent (Asd)",
  "other" : "1842.87",
  "regular" : "98538.32",
  "name" : "Adario,Anthony J",
  "retro" : "0.00",
  "department_name" : "ASD Human Resources",
  "overtime" : "0.00",
  "quinn" : "0.00"
```


Output structure: (Sample)


```
"name" : "job&earn2014",
"children" : [ {
   "name" : "ASD Human Resources",
   "children" : [ {
      "name" : "Supvising Claims Agent (Asd)",
      "children" : [] } {
         "department_name" : "ASD Human Resources",
"total_earnings" : "100381.19", "title" : "Supvising Claims Agent
(Asd)", "name" : "Adario,Anthony J"}
```

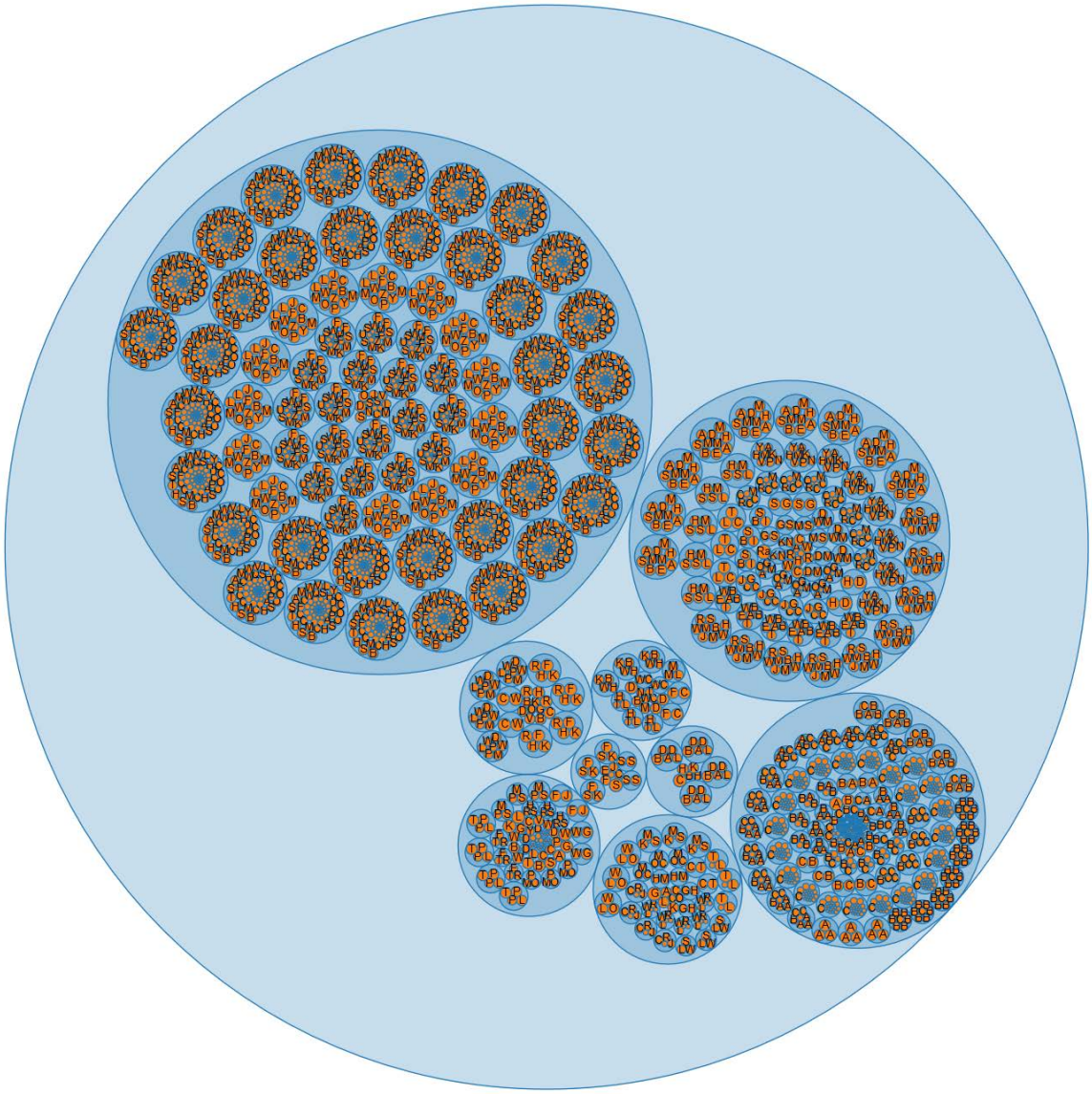

Increasing Earning: a dataset that combine Employee Earnings Report 2012 – 2014 and only keep some information of employers who have increasing earnings in the past three years.

In this project I used PyMongo which allows me to deal with datasets stored in MongoDB. I have never used MongoDB before and I found it is a good way to get familiar with MongoDB by going through the documentation of PyMongo.

This also helps me a lot when I was coding the python script. I didn't use any completed algorithm, instead I only use methods from PyMongo documentation and they work well.

I used d3 library to finish the virtualization of the datasets. I have read some guidelines and realize the virtualization of datasets is actually the virtualization of data structure. This brings me the idea to show the structure of employments in Boston. There is a performance limit which will slow down the webpage if I have input too many data.

I have made two virtualizations, the first one shows the structure of employments in Boston.



In this graph, each orange dot is the information of a person and the size of the dot shows the total earnings of that person in this year. The biggest blue circle represents the year which is 2014 in this graph. If I have input multiple reports of different years there will be multiple circles in the graph. The second biggest blue circle are different departments. Inside them, we have

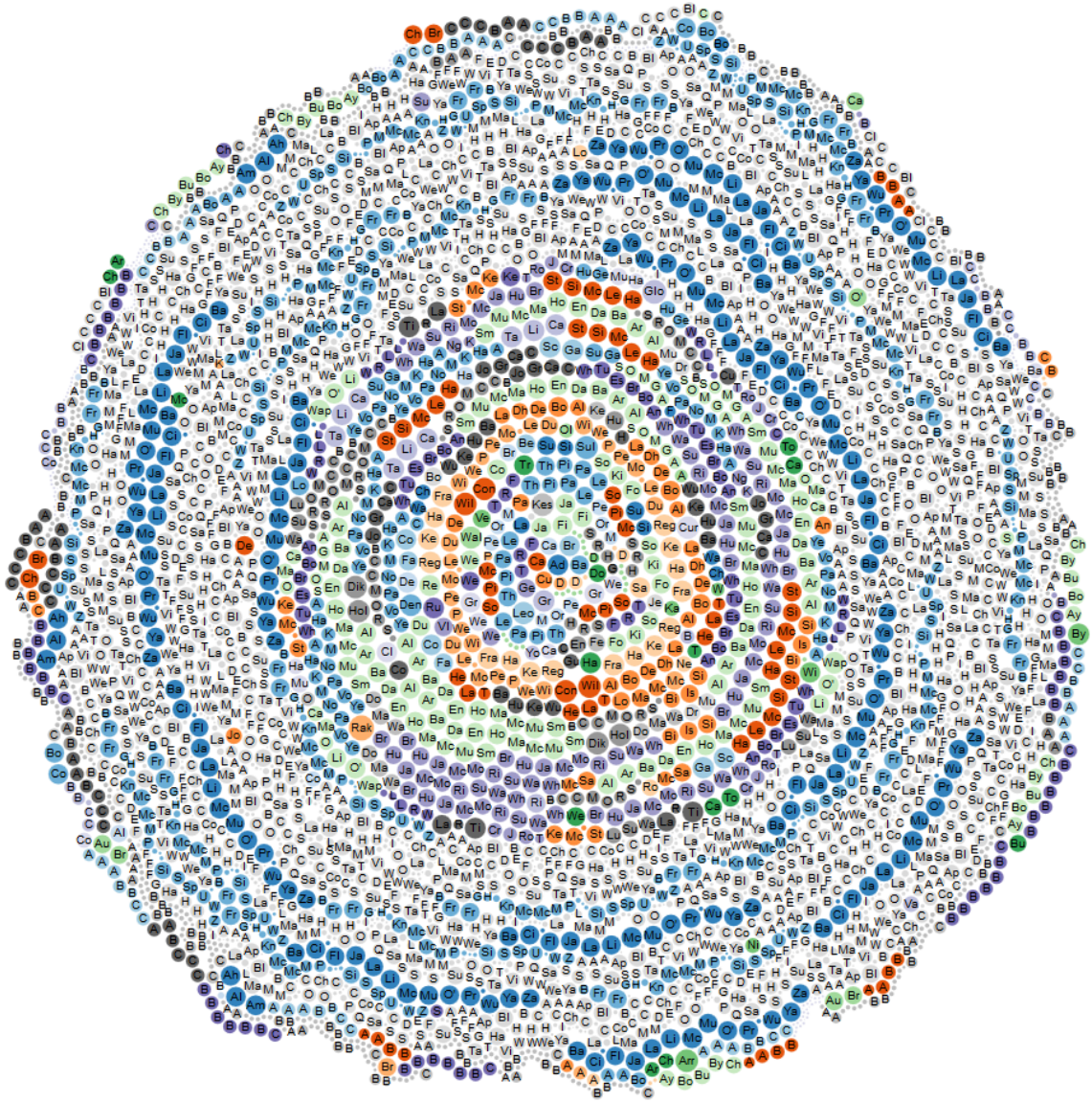
smaller circles which are various positions under these departments. From this graph we can easily see the size of different departments and how different people working as various positon in these departments.

This graph will give viewer a more directly looking about how each department are organized.

Furthermore, if we move mouse over one of the dot, we can see detailed information of that person. For example:

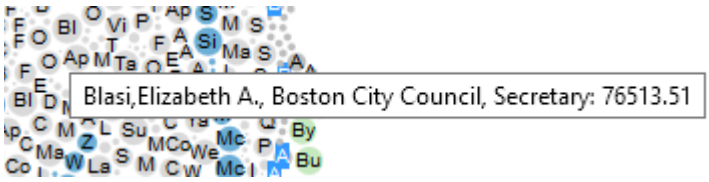
It will show the name, department name, title and total earning of the person in this year.

The second virtualization I have created is about job and positions which have increasing total earning during the past three years.



Different from the first one, this is mainly focusing on finding data with similar characteristic.

In this graph, dots with same color represent people work in the same department with same titles. We can also move the mouse over each dot to see detailed information.



In this graph, the color with most dots is gray which is Secretary under Boston City Council. Based on this we can assume that Boston City Council is looking for more secretary or it is a good time to get a job as secretary in Boston City Council.

Unfortunately, due to a performance limit, I have input about 500-600 data in the two graphs above. They can only show departments starting with letter A and B. In the feature, if I had time, I wish I can implement a function that hide the most detailed dots (for example, the personal information) and show a clearer graph with title as the smallest units (currently, person is the smallest unit). In this way I can load more data into the graph without slowing down the webpage. To avoid losing details, I can click on the dots of titles or department to see the structure which is basically a combination of two graphs.

PS1. I have included the post in the repo just in case if you want to see it. However the poster is outdated. I have conclude most of the things I want to present in this report, so it is not necessary to read the poster again.

PS2. I have also included both virtualization in the repo, department.html is the first graph and increasing.html is the second graph.