

**indeedoor :**

a better way to find a job

Team Tufte

Jade Bailey-Assam

Lucy Drotning

Christine Lee

Shruti Pandey

Janet Prumachuk

# Motivation



what

job title, keywords or company name

where

city, state or zip code

[Advanced Job Search](#)



[Upload your resume](#) - Let employers find you

# Motivation

[Find Jobs](#) [Find Resumes](#) [Employers / Post Job](#)

[Upload your resume](#) [Sign in](#)



## Advanced Job Search

### Find Jobs

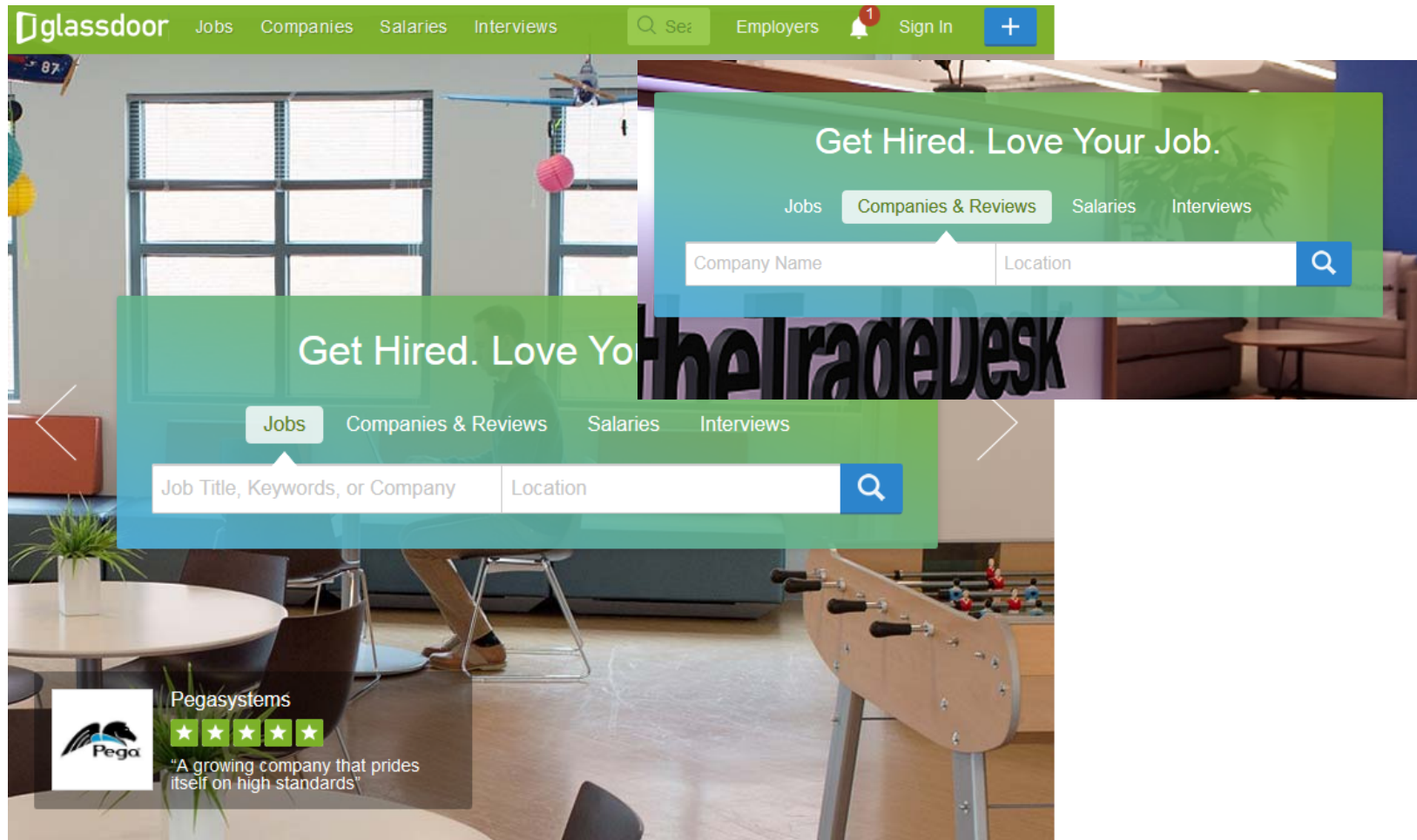
With <b>all</b> of these words	<input type="text"/>
With the <b>exact phrase</b>	<input type="text"/>
With <b>at least one</b> of these words	<input type="text"/>
With <b>none</b> of these words	<input type="text"/>
With these words in the <b>title</b>	<input type="text"/>
From this <b>company</b>	<input type="text"/>
Show jobs of type	<input type="text" value="All job types"/>
Show jobs from	<input type="text" value="All web sites"/>
	<input type="checkbox"/> Exclude staffing agencies
Salary estimate	<input type="text"/> per year
	<small>\$50,000 or \$40K-\$90K</small>

### Where and When

Location	<input type="text" value="within 25 miles of"/>	<input type="text" value="new york, ny"/>	(city, state, or zip)
Age - Jobs published	<input type="text" value="anytime"/>		
Display	<input type="text" value="10"/>	results per page, sorted by	<input type="text" value="relevance"/>

Find Jobs

# Motivation



## Your Next Career Move Starts Here



### Search Millions of Job Listings

Glassdoor has more jobs than any other job site



**See Real Employee Salaries**  
See anonymous salary details for any job or company



### Read Reviews from Employees

See reviews from employees to help you decide if you want to work in HR or Recruiting?

# Our contribution

- Both sites contain valuable, under-leveraged data
- Our goal is to give the job seeker more information as part of their job search process:
  - How are the companies rated by industry, overall and by category (Janet, Lucy)
    - Learn more about a particular set of companies (based on some user inputs), including number of reviews, number of positions open...
  - Where are the jobs located (Christine)
  - Differences in jobs by region & regression (Shruti)
  - Leverage job postings to tailor your resume (Jade)

# Starting medium

File Edit View History Bookmarks Tools Help

Lucy's fil... http...ENGs http...ENGs http...ENGs http...ENGs jcp1016/e... indeedoor EDAV 4701...

127.0.0.1:7131

r help grid.arrange c

New York, New York (... 12 Google Calendar Gmail Home - The Chronicle... Planning and Institutio... The Mac Weekly | Mac...

Edit Post to Blog

## EDAV 4701-->Find an Employer

Find the job of your dreams

**Choose a Location**

New York

**Range of Overall Review:**

2 5

### Companies that fit the bill

The image displays three bar charts side-by-side, each representing a different metric for companies in New York. The first chart, 'GD Overall Rating', shows the average rating from Glassdoor, with a scale from 0.0 to 5.0. The second chart, 'Glassdoor Reviews per 100 people', shows the number of reviews per 100 employees, with a scale from 0 to 250. The third chart, 'Indeed Job Postings (Count)', shows the number of job postings, with a scale from 0 to 100. Each chart lists companies on the y-axis, with the top 20 companies highlighted in red, blue, and green respectively.

Company	GD Overall Rating	Glassdoor Reviews per 100 people	Indeed Job Postings (Count)
Google	4.5	150	100
The McGraw-Hill Companies	4.4	140	90
MetLife	4.3	130	80
Four Seasons	4.2	120	70
Facebook	4.1	110	60
Google	4.0	100	50
Worthington	3.9	90	40
Twitter	3.8	80	30
Endava	3.7	70	20
Memorial Sloan-Kettering	3.6	60	10
SAP	3.5	50	0
Palantir Technologies	3.4	40	0
Workbridge Associates	3.3	30	0
Spotify	3.2	20	0
Capgemini	3.1	10	0
Medidata Solutions	3.0	0	0
Sony Music	2.9	0	0
Pyroptics	2.8	0	0
Medline	2.7	0	0
Accurate	2.6	0	0
Deloitte	2.5	0	0
American Express	2.4	0	0
Morgan Stanley	2.3	0	0
United	2.2	0	0
Capital One	2.1	0	0
Zurich North America	2.0	0	0
AIG	1.9	0	0
City of New York	1.8	0	0
Audience Science	1.7	0	0
World Council of Insurance	1.6	0	0
AppRover	1.5	0	0
SUNY Downstate	1.4	0	0
That's Nice LLC	1.3	0	0
Humana	1.2	0	0
Talx-CRIP	1.1	0	0
Yodle	1.0	0	0
Realtor & Partner	0.9	0	0
World Fager	0.8	0	0
Kelly Services	0.7	0	0
Finco	0.6	0	0
Morgan & Partner	0.5	0	0
Wynn Software	0.4	0	0
ASPCA	0.3	0	0
Blue Apron	0.2	0	0
MetLife	0.1	0	0
MSG Health	0.0	0	0
Success Academy	0.0	0	0
3D Systems	0.0	0	0
WU	0.0	0	0
Bank New	0.0	0	0
Over Jones	0.0	0	0
Kaplan	0.0	0	0
SWI Insurance	0.0	0	0
SWI	0.0	0	0
Guaranteed Income	0.0	0	0
MSG Entertainment	0.0	0	0
SWI Health	0.0	0	0
SWI NYC	0.0	0	0
SWI Technology	0.0	0	0
SWI Research	0.0	0	0
Intermap Network	0.0	0	0

## R Code to generate comparison charts (all black bars – very New York data scientist)

```
dscomprev <- read.csv("Workbook1.csv")
```

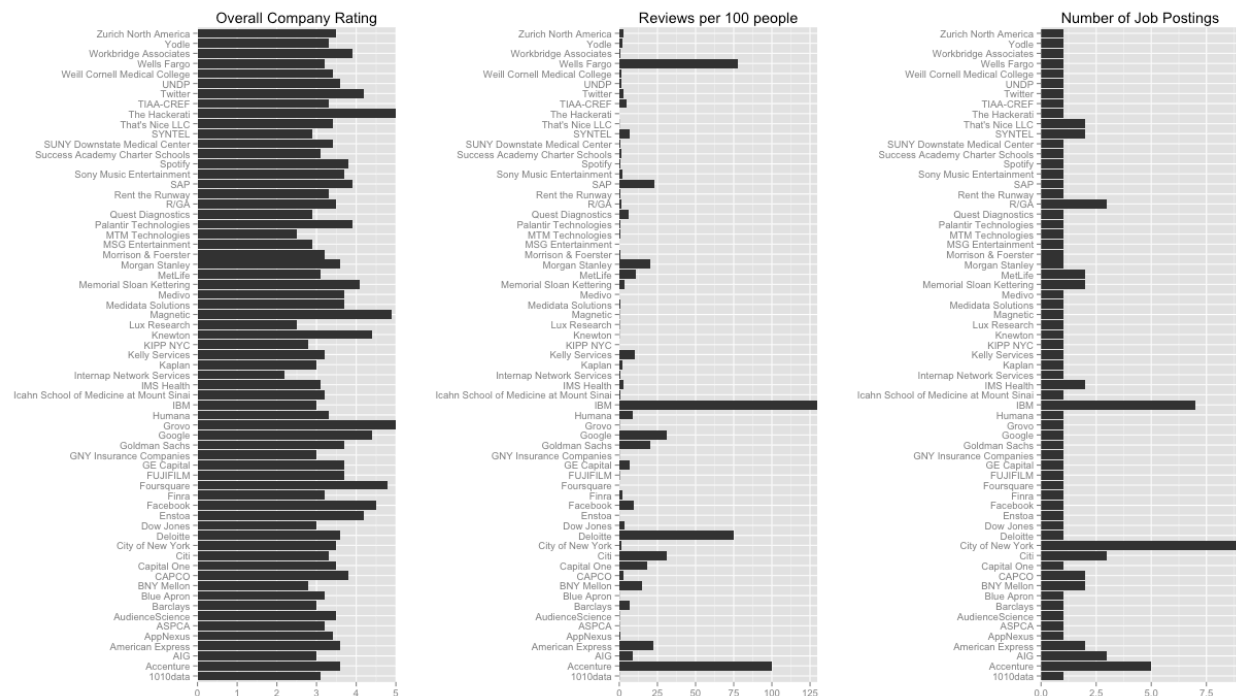
```
library(ggplot2)
library(gridExtra)
```

```
x <- ggplot(dscomprev, aes(x=Company.Name, y=Indeed.Postings)) +
  geom_bar(stat="identity") +
  coord_flip() +
  ggtitle("Number of Job Postings") +
  theme(axis.title.x = element_blank(), axis.title.y = element_blank())
x1 <- x + scale_y_continuous(expand = c(0, 0))
```

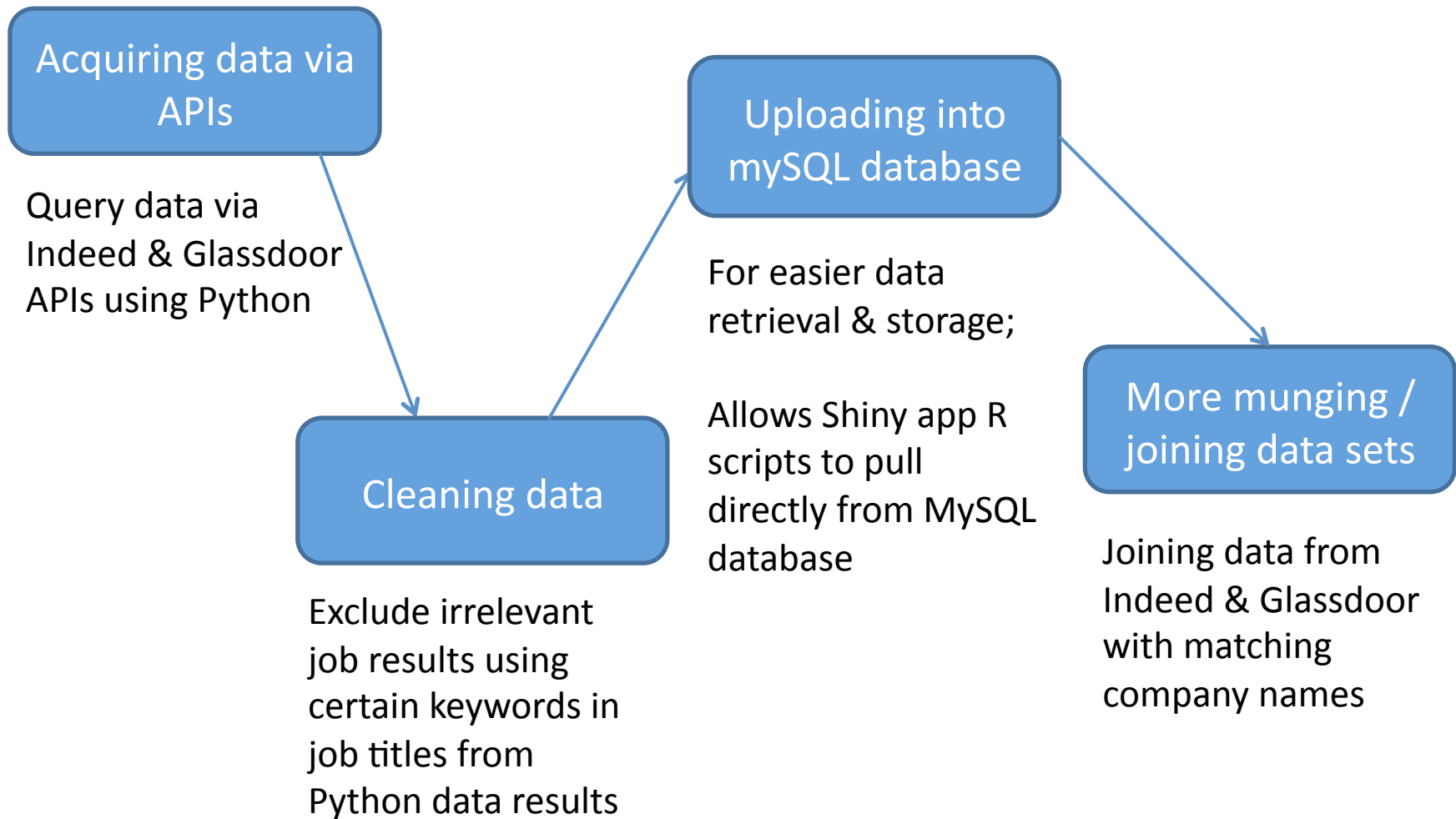
```
y <- ggplot(dscomprev, aes(x=Company.Name, y=Reviews)) +
  geom_bar(stat="identity") +
  coord_flip() +
  ggtitle("Reviews per 100 people") +
  theme(axis.title.x = element_blank(), axis.title.y = element_blank())
y1 <- y + scale_y_continuous(expand = c(0, 0))
```

```
z <- ggplot(dscomprev, aes(x=Company.Name, y=Overall.Rating)) +
  geom_bar(stat="identity") +
  coord_flip() +
  ggtitle("Overall Company Rating") +
  theme(axis.title.x = element_blank(), axis.title.y = element_blank())
z1 <- z + scale_y_continuous(expand = c(0, 0))
```

```
grid.arrange(z1, y1, x1, ncol=3)
```



# Our data pipeline





# Data acquisition & preparation

## Indeed.com jobs data

### Scope:

- Jobs with query term “data + scientist”
- Job results within 40 miles or less of 12 cities in the U.S.

### Challenges in cleaning data:

- Created conditions in Python script that filtered out irrelevant job titles
- But positions that have term “data scientist” in description still show up
- Meanwhile, we also cannot simply eliminate seemingly “irrelevant” job titles as some of these jobs are actually data science positions with a different name

## Glassdoor company data

### Scope:

- Company Name
- Location, Ratings

### Challenges in cleaning data:

- Minimal data cleaning required
- Ignored rating text fields

The API call only shows us the job snippet, which isn't enough to mine. To get enough data for a POC, we'll extract the URL from the indeed job posting, scrape this text and mine it.

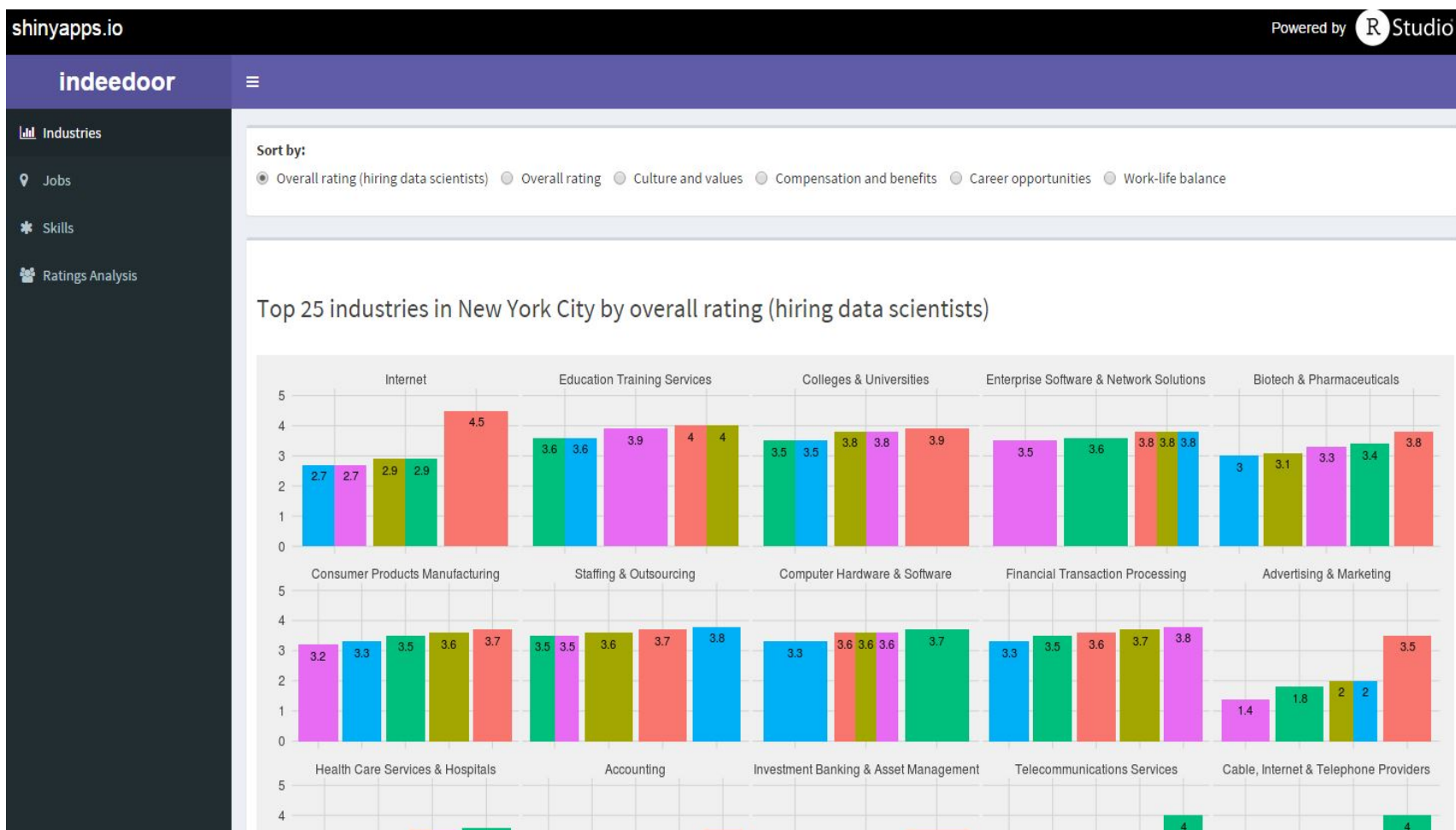
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<response version="2">
  <query>data scientist</query>
  <location>new, york</location>
  <dupefilter>true</dupefilter>
  <highlight>>false</highlight>
  <totalresults>1147</totalresults>
  <start>1</start>
  <end>10</end>
  <pageNumber>0</pageNumber>
  ▼<results>
    ▼<result>
      <jobtitle>Data Scientist</jobtitle>
      <company>The College Board</company>
      <city>New York</city>
      <state>NY</state>
      <country>US</country>
      <formattedLocation>New York, NY</formattedLocation>
      <source>The College Board</source>
      <date>Mon, 27 Apr 2015 15:55:37 GMT</date>
      ▼<snippet>
        Experience with descriptive statistics and data visualization in R and Tableau. Create insights from existing data, and drive the collection of new data through...
      </snippet>
      ▼<url>
        http://www.indeed.com/viewjob?jk=aa482c194e06d589&qd=43t9H8wdm9_p3y0mSnUcn3G_LekVwgh1sg0a_coxuzer1EA6V7HKKpskzdwznzVRwCNFapReE6dZrtlp5SIE6ou9Bek1LMgTs8p5EmYwvc5ULB2pHYt-YP2NYagtBg9Lp&indpubnum=4751269202013823&atk=19k10ebv75ucse6f
      </url>
      <onmousedown>indeed_clk(this, '4963');</onmousedown>
      <latitude>40.71154</latitude>
      <longitude>-74.00549</longitude>
      <jobkey>aa482c194e06d589</jobkey>
      <sponsored>>false</sponsored>
      <expired>>false</expired>
      <indeedApply>>false</indeedApply>
      <formattedLocationFull>New York, NY</formattedLocationFull>
      <formattedRelativeTime>1 day ago</formattedRelativeTime>
    </result>
```

**Trim URL, add to list to be scraped.**  
**API call gave 10 URLs to work with for POC**

**URL from API to grab for full job description**

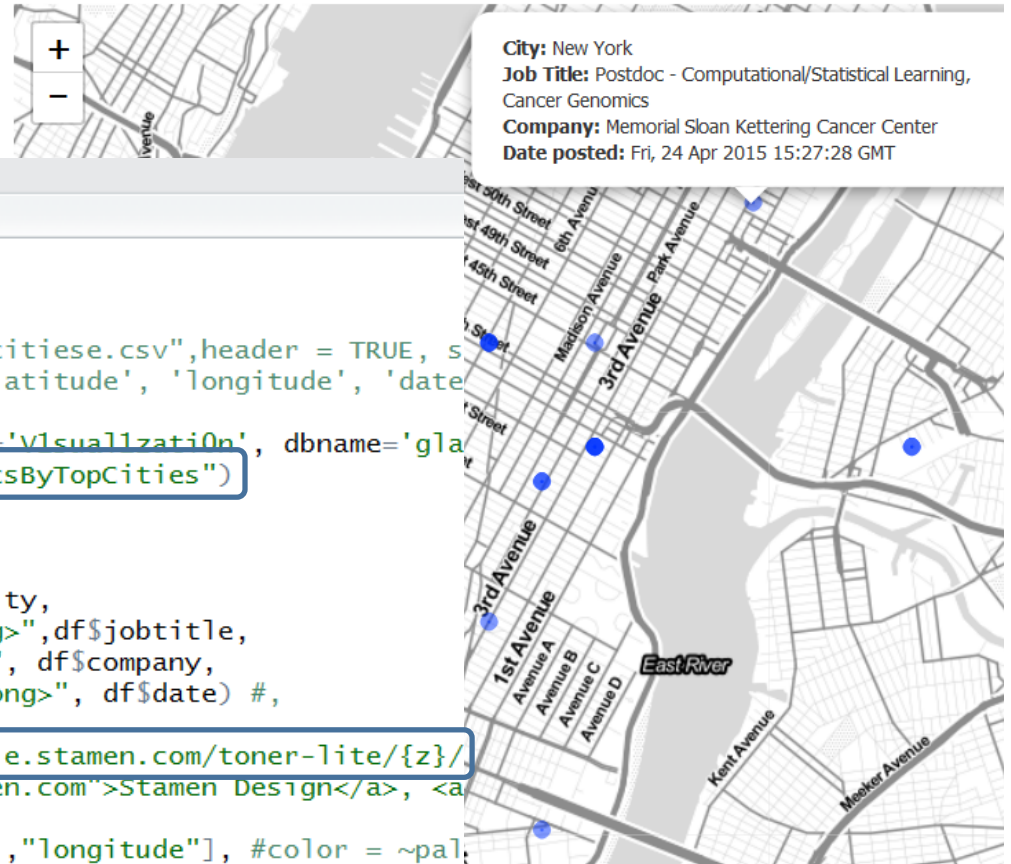
# The indeedoor app



# Shiny code highlight – jobs map app

Dots on the map represent job postings and their locations from a search Indeed using query term "data scientist":

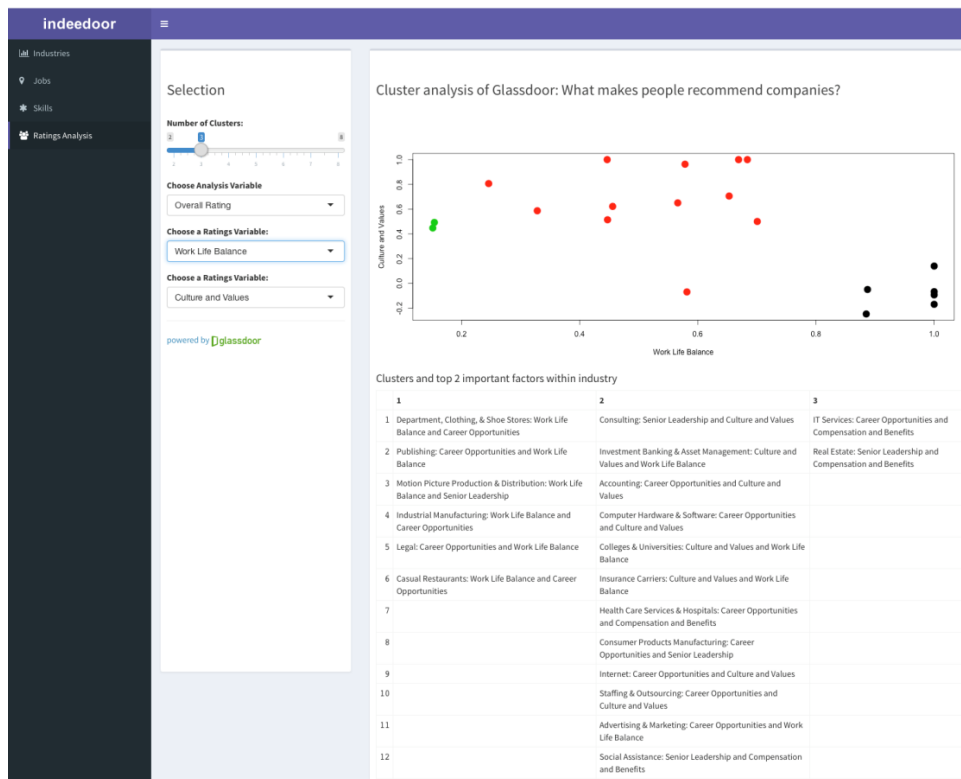
Please click on job location circle for more information for each job posting



```
shinyServer(function(input, output) {  
  library(rgdal)  
  #indeeddata<- read.csv("./data/Newindeed_datasci_topcities.csv",header = TRUE, s  
  # col.names=c('city','state','snippet', 'latitude', 'longitude', 'date'  
  mydb = dbConnect(MySQL(), user='STATw4701', password='V1sual1zati0n', dbname='gla  
  rs <- dbSendQuery(mydb, "select * from IndeedAnalyticsByTopCities")  
  indeeddata <- fetch(rs, n=-1)  
  df<-data.frame(indeeddata)  
  
  scrape_pop <- paste0("<strong>City: </strong>", df$city,  
    "<br><strong>Job Title: </strong>", df$jobtitle,  
    "<br><strong>Company: </strong>", df$company,  
    "<br><strong>Date posted: </strong>", df$date) #,  
  
  m2 <- leaflet(data = df) %>% addTiles('http://{s}.tile.stamen.com/toner-lite/{z}/  
    attribution = 'Map tiles by <a href="http://stamen.com">Stamen Design</a>, <a  
    setView(-73.9983273, 40.7471983, zoom = 12) %>%  
    addCircles(lat = ~ df[, "latitude"], lng = ~ df[, "longitude"], #color = ~pal,  
      fillOpacity = .5,  
      radius = 8,  
      popup= scrape_pop,  
      weight = 9)  
  
  output$ClistMap = renderLeaflet(m2)  
})
```

```
library(leaflet)  
library(ggplot2)  
library(maps)  
library(RMySQL)
```

# Regression: Screenshot and Code Snippet



```
getRegressionAnalysis <- memoise(function(variable1, variable2, k, clustertype){
  #con <- getConnection()
  #rs = dbSendQuery(mydb, "select * from CompanyRatings")
  #data = fetch(rs, n=-1)
  data <- master_gd_data ## already have this in a data frame
  print(variable1)
  print(variable2)
  data <- subset(data, data$employers.industry != "")
  industries <-unique(data$employers.industry)
  largedata<-subset(data, data$employers.numberOfRatings > 40)
  clusteringdata<-array()
  nameofindustry<-vector()
  mostimportant<-array()
  j <- 0
  for (i in industries)
  {
    fdata <- subset(largedata, data$employers.industry == i)
    if (nrow(fdata) > 40)
    {
      print (paste0("Industry is ", i, " and the number of comp
      id <- fdata$employers.id
      name <- fdata$employers.name
      website <- fdata$employers.website
      industry <- fdata$employers.industry

      overallRating <- fdata$employers.overallRating
      ratingDescription <- fdata$employers.ratingDescription
      cultureAndValuesRating <- fdata$employers.cultureAndValue
      seniorLeadershipRating <- fdata$employers.seniorLeadershi
      compensationAndBenefitsRating <- fdata$employers.compensc
      careerOpportunitiesRating <- fdata$employers.careerOpport
      workLifeBalanceRating <- fdata$employers.workLifeBalance
      recommendToFriendRating <- fdata$employers.recommendToFri
      pctApprove <- fdata$employers.ceo.pctApprove

      employer <- data.frame(id, name, website, industry, overc
    }
  }
})
```

# Skills: Screenshot and Code Snippet

indeedoor

Industries

Jobs

Skills

Ratings Analysis

Selection

Choose a city:

NYC

Austin

Boston

Chicago

Houston

NYC

San Francisco

San Jose

jobs by Indeed

Frequent words found in Data Science job postings



```
getTermMatrix <- memoise(function(dbCity) {  
  ##con <- getConnection() ## this was giving errors and corrupting the co  
  rs = dbSendQuery(mydb, paste0("select * from IndeedAnalyticsByTopCities w  
  jobData = fetch(rs, n=1)  
  #dbDisconnect(con)  
  
  # Read in data as dataFrame  
  # reading only the text snippets  
  x<-data.frame(v1=jobData$snippet)  
  x<-subset(x,x$v1 != "")  
  docs<- Corpus(DataframeSource(x))  
  
  # now for some cleansups  
  docs <- tm_map(docs, content_transformer(tolower))  
  docs <- tm_map(docs, removeNumbers)  
  docs <- tm_map(docs, removePunctuation)  
  docs <- tm_map(docs, removeWords, stopwords("english"))  
  docs <- tm_map(docs, removeWords, c("experience", "will", "data", "analyt  
  docs <- tm_map(docs, stripWhitespace)  
  # docs <- tm_map(docs, stemDocument)  
  
  # create the Document Term Matrix and compute the frequencies  
  dtm <- DocumentTermMatrix(docs)  
  m <- as.matrix(dtm)  
  sort(colSums(m), decreasing = TRUE)  
  #freqs <- colSums(as.matrix(dtm))  
  
})  
getVariables <-(function(choice)
```

# Text mining code

-Try using rvest() to scrape URLs from R. This is not successful, use import.io to scrape. Merge and organize in old, faithful friend Excel.

-Use text mining tm() package in R to compile stop words and do word stemming. Code below

```
filterlistnew <- removeWords(filterlist, stopwords, c ("data", "hour", "federal", "within", "hagan",  
"yeeldr", "bank", "get", "we're", "ricci", "top", "ability", "like", "also", "skill", "working", "job", "trend",  
"new", "york", "ny", "company", "days", "work", "ago", "one", "opportunity", "experience",  
"scientist", "education", "city", "program", "state", "use", "capital", "apply", "etc", "contact", "equal",  
public", "find", "ll", "employment", "post", "action", "title", "resume", "inc", "save", "times", "well", "fo  
rum", "require", "well", "terms", "cookie", "smart", "upload", "protect", "resumesemployer", "2015",  
"search", "keyword", "high", "responsibility", "zip", "jobsfind", "indeed", "privacy", "help", "sign", "ind  
eed.com", "affirmative", "veteran", "review", "policy", "question", "email", "world", "disability", "inclu  
ding", "friend", "demonstrate", "student", "candidate", "us", "real", "knewton", "include", "view", "pri  
ce", "employee", "background", "enstoa", "benefit", "status", "environment"))
```

```
filterlistnew <- wordStem(c ("machine", "learning", 'machine learning', "statistic", "statistics",  
'statistics', "communicate", "communication", 'communication'))
```



# Leverage job postings to tailor your resume

These ten “specific skills” that might be helpful to have on your resume

Ex: Proficiency in Python, SQL, Hadoop, Tableau, MapReduce

```
## Source: local data frame [10 x 3]
##
##      Word Frequency      Group
## 1      python         11 specific skills
## 2         sql          8 specific skills
## 3   language          7 specific skills
## 4     hadoop          7 specific skills
## 5 production          7 specific skills
## 6   technique          7 specific skills
## 7        java          6 specific skills
## 8   mapreduce          5 specific skills
## 9     cluster          4 specific skills
## 10    tableau          4 specific skills
```

These six “coursework” words that might be helpful to include on your resume

Ex: Relevant Engineering coursework in: statistics, mathematics, machine learning, algorithms, computer science

```
## Source: local data frame [6 x 3]
##
##      Word Frequency      Group
## 1    statistics         18 relevant coursework
## 2   mathematics         17 relevant coursework
## 3 machine learning         15 relevant coursework
## 4     algorithm         13 relevant coursework
## 5     engineer          12 relevant coursework
## 6 computer science          6 relevant coursework
```



# Continued....

## Rest of groups published on RPubS

These are the top “buzzwords” that might be helpful to have on your resume

Ex: Identify insights to benchmark digital knowledge through use of big data and advanced analytics

```
## Source: local data frame [16 x 3]
##
##      Word Frequency      Group
## 1  analytics        16 buzzwords
## 2   advanced        11 buzzwords
## 3    build          9 buzzwords
## 4  knowledge         9 buzzwords
## 5   digital         8 buzzwords
## 6 quantitative       7 buzzwords
## 7   research         6 buzzwords
## 8   big data         6 buzzwords
## 9  investigate       5 buzzwords
## 10   insight         5 buzzwords
## 11   expert          5 buzzwords
## 12  identify         5 buzzwords
## 13   predict         5 buzzwords
## 14   impact          4 buzzwords
## 15  benchmark         4 buzzwords
## 16 understand        4 buzzwords
```

# Shiny apps integration

- What went well
- Key challenges...
- Lessons learned

# Challenges and Next Steps

- Disparate, incomplete data
  - Difficulty merging across jobs and companies datasets
  - Is Shiny the best tool?
- 
- Expand dataset to include more jobs, more locations, more company and position specific detail
  - Continue to develop presentation & connection between the parts
  - Knock on the Glassdoor with our app....Indeed!

# Where to find our work

- <http://www.indeedoor.com>
- <https://github.com/jcp1016/edav-team-project/indeedoor>
- <http://rpubs.com/jadeemily/textmining>
- <http://rpubs.com/lucy/78463>