cybercoders

Yao Yao

March 16, 2017

Data Science Profiles Analysis

Introduction

CyberCoders is a website for technical job postings. We will user R to understand the market for data science based on the job postings on https://www.cybercoders.com

Data Collection

We will use XML (https://cran.r-project.org/web/packages/XML/index.html). First, we will write functions to parse the website and the collect the required data. Following chunk of code shows how to search on the website using R. *RCurl::getForm* submits our search query to the website. Also, after taking a look at the html code for the website (*View page source* in the browser), let's also extract the hyperlinks for each of the job postings on the webpage.

Load the libraries that are used for data collection.

```
library(XML)
library(RCurl)
```

```
## Loading required package: bitops
```

```
library(tm)
```

```
## Warning: package 'tm' was built under R version 3.3.3
```

```
## Loading required package: NLP
```

```
library(wordcloud)
```

```
## Warning: package 'wordcloud' was built under R version 3.3.3
```

```
## Loading required package: RColorBrewer
```

Submit search query to website

```
# Extract the search web page
txt = getForm("https://www.cybercoders.com/search/", searchterms = '"Data Scientist"',
              searchlocation = "",  newsearch = "true", sorttype = "")
doc <- htmlParse(txt, asText = TRUE)
# Extract links for each job posting
links <- getNodeSet(doc, "//div[@class = 'job-title']/a/@href")
joblinks <- getRelativeURL(as.character(links), "https://www.cybercoders.com/search/")

print(length(joblinks))
```

```
## [1] 20
```

There are 20 job postings on the first web page for the query of Data Scientist. After looking the page source for the job posting, we can extract the information of the posting like the job description, posting date, skills required and salary range. Let's write seperate functions to extract each of these and test them.

Extract attributes of job posting

First, let's extract the job description and test it with the first link in our *joblinks* variable. The function will extract the text, remove punctuations, convert to lower case and remove stop words. It will return a list of words that are present in the text. This will be useful in understanding the key words used by job posters who are looking for Data Scientists.

```
# Remove stop words
removeStopWords = function(x, stopWords = stopwords(kind='en'))
{
  if(is.character(x))
    setdiff(x, stopWords)
  else if(is.list(x))
    lapply(x, removeStopWords, stopWords)
  else
    x
}


# Extract text of job description
cy.getJobWords = function(doc)
{
  nodes = getNodeSet(doc, "//div[@class='job-details']/
                     div[@data-section]") # Match node criteria and extract corresponding text

  if(length(nodes) == 0)
    nodes = getNodeSet(doc, "//div[@class='job-details']//p")

  if(length(nodes) == 0)
    warning("did not find any nodes for the free form text in ",
            docName(doc))
  # Split the text into individual words and remove punctuations
  words = lapply(nodes,
                 function(x)strsplit(xmlValue(x), "[[:space:][:punct:]]+"))
  words_lower <- removeNumbers(tolower(unlist(words))) # Conver to lower case and remove numbers
  return(removeStopWords(words_lower))
}

job_doc <- htmlParse(getURLContent(joblinks[1])) # HTML page
jd <- cy.getJobWords(doc=job_doc)
jd
```

```
##   [1] "data"           "scientist"      "business"       "analyst"
##   [5] "software"       "engineer"       "phd"            "life"
##   [9] "sciences"       "re"             "interested"     "solving"
##  [13] "complex"        "problems"       "please"         "read"
##  [17] "global"         "analytics"      "services"       "consulting"
##  [21] "company"        "works"          "world<U+0092>s"        "leading"
##  [25] "organizations"  "solve"          "real"           "world"
```

```
##  [29] "seeking"        "talented"       "developer"      "strong"
##  [33] "academic"       "commercial"     "background"     "sector"
##  [37] "role"           "will"           "helping"        "us"
##  [41] "add"            "value"          "clients"        "difficult"
##  [45] "challenging"    "face"           "offer"          "include"
##  [49] "informatics"    "expertise"      "consultancy"    "exceptional"
##  [53] "individual"     "join"           "work"           "across"
##  [57] "areas"          "exciting"       "projects"       "illustrious"
##  [61] "customers"      "varied"         "can"            "span"
##  [65] "range"          "activities"     ""               "working"
##  [69] "closely"        "help"           "identify"       "define"
##  [73] "requirements"   "understand"     "make"           "use"
##  [77] "improve"        "processes"      "inform"         "decision"
##  [81] "making"         "design"         "develop"        "tailor"
##  [85] "made"           "solutions"      "tools"          "getting"
##  [89] "involved"       "stages"         "development"    "lifecycle"
##  [93] "successful"     "applicant"      "pharmaceutical" "biotech"
##  [97] "boston"         "new"            "jersey"         "opportunities"
## [101] "industries"     "locations"      "may"            "arise"
## [105] "future"         "bs"             "min"            "gpa"
## [109] "ms"             "science"        "engineering"    "math"
## [113] "focus"          "degree"         "biology"        "chemistry"
## [117] "bioinformatics" "similar"        "ideal"          "understanding"
## [121] "ability"        "interpret"      "large"          "volumes"
## [125] "draw"           "conclusions"    "experience"     "one"
## [129] "following"      "java"           "r"              "python"
## [133] "c"              "industry"       "e"              "g"
## [137] "internship"     "project"        "advantage"      "excellent"
## [141] "written"        "verbal"         "communication"  "skills"
## [145] "explain"        "technical"      "colleagues"     "backgrounds"
## [149] "careers"        "companies"      "forefront"      "technology"
## [153] "staff"          "high"           "achieving"      "problem"
## [157] "solvers"        "half"           "phds"           "enjoy"
## [161] "like"           "minded"         "intelligent"    "create"
## [165] "deliver"        "difference"     "developing"     "smarter"
## [169] "drug"           "trials"         "controlling"    "orbit"
## [173] "attitude"       "satellites"     "well"           "competitive"
## [177] "salary"         "benefits"       "package"        "also"
## [181] "prospects"      "growing"        "abilities"      "leadership"
## [185] "tailored"       "career"         "based"          "aspirations"
## [189] "apply"          "today"
```

Similarly, the following function extracts the preferred skill set listed by the job poster.

```
# Extract skill set
cy.getSkillList = function(doc)
{
  lis = getNodeSet(doc, "//div[@class = 'skills-section']//
                   li[@class = 'skill-item']//
                   span[@class = 'skill-name']")

  sapply(lis, xmlValue)
}
skill_set <- cy.getSkillList(job_doc)
skill_set
```

```
## [1] "Data Analytics"
## [2] "Informatics"
## [3] "Life Sciences . Pharmaceutical Industry"
## [4] "Java"
## [5] "Python"
```

The following function extracts the basic job information like location, salary range and the date of the posting.

```
# Trim leading and trailing white spaces
trim <- function (x) gsub("^\\s+|\\s+$", "", x)

# Extract location and salary
cy.getLocationSalary = function(doc)
{
  ans <- xpathSApply(doc, "//div[@class = 'job-info-main'][1]/div",xmlValue)
  ans <- lapply(ans, trim)
  loc <- ans[[1]]
  salary <- ans[[2]]
  post_date <-  as.Date(strsplit(ans[[3]], " ")[[1]][2], format = "%m/%d/%Y")
  job_info <- list(loc, salary, post_date)
  names(job_info) <- c("Location", "Salary", "Post_Date")
  return(job_info)
}

job_info <- cy.getLocationSalary(job_doc)
job_info
```

```
## $Location
## [1] "Newton, MA"
##
## $Salary
## [1] "Full-time $100k - $130k"
##
## $Post_Date
## [1] "2017-02-23"
```

Great! Now we can use this functions to extract all the job postings on one web page.

```
# Extract the post
cy.readPost = function(u, stopWords = StopWords, doc = htmlParse(getURLContent(u)))
{
  ans = list(words = cy.getJobWords(doc),
             skills = cy.getSkillList(doc))
  o = cy.getLocationSalary(doc)
  ans[names(o)] = o
  ans
}

posts <- lapply(joblinks,cy.readPost)
posts[1]
```

```
## $`/data-scientist-job-266569`
## $`/data-scientist-job-266569`$words
##   [1] "data"           "scientist"      "business"       "analyst"
##   [5] "software"       "engineer"       "phd"            "life"
##   [9] "sciences"       "re"             "interested"     "solving"
```

```
##  [13] "complex"        "problems"       "please"         "read"
##  [17] "global"         "analytics"      "services"       "consulting"
##  [21] "company"        "works"          "world<U+0092>s"        "leading"
##  [25] "organizations"  "solve"          "real"           "world"
##  [29] "seeking"        "talented"       "developer"      "strong"
##  [33] "academic"       "commercial"     "background"     "sector"
##  [37] "role"           "will"           "helping"        "us"
##  [41] "add"            "value"          "clients"        "difficult"
##  [45] "challenging"    "face"           "offer"          "include"
##  [49] "informatics"    "expertise"      "consultancy"    "exceptional"
##  [53] "individual"     "join"           "work"           "across"
##  [57] "areas"          "exciting"       "projects"       "illustrious"
##  [61] "customers"      "varied"         "can"            "span"
##  [65] "range"          "activities"     ""               "working"
##  [69] "closely"        "help"           "identify"       "define"
##  [73] "requirements"   "understand"     "make"           "use"
##  [77] "improve"        "processes"      "inform"         "decision"
##  [81] "making"         "design"         "develop"        "tailor"
##  [85] "made"           "solutions"      "tools"          "getting"
##  [89] "involved"       "stages"         "development"    "lifecycle"
##  [93] "successful"     "applicant"      "pharmaceutical" "biotech"
##  [97] "boston"         "new"            "jersey"         "opportunities"
## [101] "industries"     "locations"      "may"            "arise"
## [105] "future"         "bs"             "min"            "gpa"
## [109] "ms"             "science"        "engineering"    "math"
## [113] "focus"          "degree"         "biology"        "chemistry"
## [117] "bioinformatics" "similar"        "ideal"          "understanding"
## [121] "ability"        "interpret"      "large"          "volumes"
## [125] "draw"           "conclusions"    "experience"     "one"
## [129] "following"      "java"           "r"              "python"
## [133] "c"              "industry"       "e"              "g"
## [137] "internship"     "project"        "advantage"      "excellent"
## [141] "written"        "verbal"         "communication"  "skills"
## [145] "explain"        "technical"      "colleagues"     "backgrounds"
## [149] "careers"        "companies"      "forefront"      "technology"
## [153] "staff"          "high"           "achieving"      "problem"
## [157] "solvers"        "half"           "phds"           "enjoy"
## [161] "like"           "minded"         "intelligent"    "create"
## [165] "deliver"        "difference"     "developing"     "smarter"
## [169] "drug"           "trials"         "controlling"    "orbit"
## [173] "attitude"       "satellites"     "well"           "competitive"
## [177] "salary"         "benefits"       "package"        "also"
## [181] "prospects"      "growing"        "abilities"      "leadership"
## [185] "tailored"       "career"         "based"          "aspirations"
## [189] "apply"          "today"
##
## $`/data-scientist-job-266569`$skills
## [1] "Data Analytics"
## [2] "Informatics"
## [3] "Life Sciences . Pharmaceutical Industry"
## [4] "Java"
## [5] "Python"
##
## $`/data-scientist-job-266569`$Location
```

```
## [1] "Newton, MA"
##
## $`/data-scientist-job-266569`$Salary
## [1] "Full-time $100k - $130k"
##
## $`/data-scientist-job-266569`$Post_Date
## [1] "2017-02-23"
```

posts variable is a named list which holds information about each of the job postings on first web page of the query. The following function puts all of the above steps in one function.

```
cy.getPostLinks = function(doc, baseURL = "https://www.cybercoders.com/search/")
{
  if(is.character(doc)) doc = htmlParse(doc)
  links = getNodeSet(doc, "//div[@class = 'job-title']/a/@href")
  getRelativeURL(as.character(links), baseURL)
}

cy.readPagePosts = function(doc, links = cy.getPostLinks(doc, baseURL),baseURL = "https://www.cybercode
{
  if(is.character(doc)) doc = htmlParse(doc)
  lapply(links, cy.readPost)
}

## Testing the function with the parsed version of the first page of results in object doc
posts = cy.readPagePosts(doc)
sapply(posts,`[[`, "Salary")
```

```
##           /data-scientist-job-266569           /data-scientist-job-326486
##            "Full-time $100k - $130k"            "Full-time $100k - $120k"
##           /data-scientist-job-332266           /data-scientist-job-174173
##            "Full-time $150k - $200k"             "Full-time $95k - $120k"
##           /data-scientist-job-326900           /data-scientist-job-336749
## "Full-time Compensation Unspecified"            "Full-time $160k - $225k"
##           /data-scientist-job-309929           /data-scientist-job-316185
##            "Full-time $150k - $200k" "Full-time Compensation Unspecified"
##           /data-scientist-job-321761           /data-scientist-job-326673
##            "Full-time $140k - $225k" "Full-time Compensation Unspecified"
##           /data-scientist-job-315448           /data-scientist-job-332391
##            "Full-time $120k - $150k" "Full-time Compensation Unspecified"
##           /data-scientist-job-333554           /data-scientist-job-312274
## "Full-time Compensation Unspecified"            "Full-time $100k - $150k"
##           /data-scientist-job-323532           /data-scientist-job-335486
## "Full-time Compensation Unspecified"            "Full-time $100k - $130k"
##           /data-scientist-job-316090           /data-scientist-job-267769
## "Full-time Compensation Unspecified"            "Full-time $100k - $175k"
##           /data-scientist-job-331774           /data-scientist-job-331926
## "Full-time Compensation Unspecified"            "Full-time $160k - $200k"
```

Now, we would like to go the next page of the search query and extract information about all of the job postings. Following function facilitates navigating to the next page and the above functions could then be used to extract the data.

```
## A function to get all pages
cy.getNextPageLink = function(doc, baseURL = docName(doc))
{
```

```
  if(is.na(baseURL))
    baseURL = "https://www.cybercoders.com/"
  link = getNodeSet(doc, "//li[@class = 'lnk-next pager-item ']/a/@href")
  if(length(link) == 0)
    return(character())
  link2 <- gsub("./", "search/",link[[1]])
  getRelativeURL(link2, baseURL)
}

# Test the above function
tmp = cy.getNextPageLink(doc, "http://www.cybercoders.com")
tmp
```

```
## [1] "http://www.cybercoders.com/search/?page=2&searchterms=%22Data%20Scientist%22&searchlocation=&new
```

All in One!

The following function combines all the functions together to automatically extract all the job postings from the website and store it in the desired format.

```
cyberCoders =
  function(query)
  {
    txt = getForm("https://www.cybercoders.com/search/",
                  searchterms = query,  searchlocation = "",
                  newsearch = "true",  sorttype = "")
    doc = htmlParse(txt)

    posts = list()
    while(TRUE) {
      posts = c(posts, cy.readPagePosts(doc))
      nextPage = cy.getNextPageLink(doc)
      if(length(nextPage) == 0)
        break

      nextPage = getURLContent(nextPage)
      doc = htmlParse(nextPage, asText = TRUE)
    }
    invisible(posts)
  }

dataScience <- cyberCoders("Data Scienctist")
skillSet = sort(table(unlist(lapply(dataScience, `[[`, "skills"))),
                decreasing = TRUE)
skillSet[skillSet >= 2]
```

```
##
##                                 Machine Learning
##                                               63
##                                           Python
##                                               61
##                                      Data Mining
##                                               30
```

```
##                                    Hadoop
##                                        29
##                                         R
##                                        25
##                                       SQL
##                                        21
##                                      Java
##                                        14
##                                     SPARK
##                                        14
##                              Data Science
##                                        13
##                               Algorithms
##                                        10
##                                     Scala
##                                         9
##                       Algorithm Development
##                                         7
##                             Deep Learning
##                                         7
##                  Machine Learning algorithms
##                                         7
##                         Statistical Modeling
##                                         7
##                              Data Analysis
##                                         6
##                             Data Scientist
##                                         6
##                                       NLP
##                                         6
##                                    pandas
##                                         6
##                                   Tableau
##                                         6
##                                       AWS
##                                         5
##                                  Big Data
##                                         5
##                                     numpy
##                                         5
##                                       PhD
##                                         5
##                                Apache Spark
##                                         4
##                           Bayesian Inference
##                                         4
##                               Data Analytics
##                                         4
##                           Data Visualization
##                                         4
##                                   mongodb
##                                         4
##                              neural networks
##                                         4
```

```
##                                Predictive Modeling
##                                                 4
##                             Python, Scala, or Java
##                                                 4
##                                          Python/R
##                                                 4
##                                               SAS
##                                                 4
##                                       scikit.learn
##                                                 4
##                                        Statistics
##                                                 4
##                                Virtual Environments
##                                                 4
##                                       BI Reporting
##                                                 3
##                                  BI Reporting Tools
##                                                 3
##                                    Business Objects
##                                                 3
##                                               C/C++
##                                                 3
##                                             Congos
##                                                 3
##                                       elasticsearch
##                                                 3
##                                                Hive
##                                                 3
##                                             Jupyter
##                                                 3
##                                               Kafka
##                                                 3
##                                           Mapreduce
##                                                 3
##                                       Micro Strategy
##                                                 3
##                                          Networking
##                                                 3
##                                               OBIEE
##                                                 3
##                                   Online advertising
##                                                 3
##                                        Random Forest
##                                                 3
##                                            Security
##                                                 3
##      Agile Framework with Atlassian (OR Similar)
##                                                 2
## Analytics Experience (Financial Industry a plus)
##                                                 2
##                           Analyzing Large Data Sets
##                                                 2
##                             Artificial Intelligence
##                                                 2
```

```
##               Basic Statistics and Data Analysis
##                                                2
##                                  Bayesian Modeling
##                                                2
##                                  Bayesian Networks
##                                                2
##                                              C++
##                                                2
##                                 Cluster Analysis
##                                                2
##                           Computer Science Degree
##                                                2
##                                  Computer Vision
##                                                2
##                                              d3js
##                                                2
##                      Data Ingestion and Processing
##                                                2
##                                  Data Manipulation
##                                                2
##                       Data science (from industry)
##                                                2
##                             Distributed Frameworks
##                                                2
##                               Financial Statements
##                                                2
##                                    Graph Analytics
##                                                2
##                                             lead
##                                                2
##                                            Linux
##                                                2
##                                     Markov Chains
##                                                2
##                                       matplotlib
##                                                2
##                       Natural Language Processing
##                                                2
##                                    NoSQL Databases
##                                                2
##                               Open Source Software
##                                                2
##                                    Personalization
##                                                2
##                                            Ph.D
##                                                2
##        Phenomenal written and oral communication
##                                                2
##                                         Principal
##                                                2
##        Prior teaching experience (not essential)
##                                                2
##                                          R/Matlab
##                                                2
```

```
##                                 Raw Data Analysis
##                                                 2
##                               Regression analysis
##                                                 2
##                                Research Scientist
##                                                 2
##                                              Ruby
##                                                 2
##                                            senior
##                                                 2
##                                     SQL Databases
##                                                 2
##                               teaching experience
##                                                 2
##       Unclean/Semi- Structured/Unstructured Data
##                                                 2
```

Analysis - Skills

A simple way to visualize all the skills with respect to their frequency is to use a word cloud.

```
wordcloud(names(skillSet),skillSet,scale = c(3,1),max.words = 25,colors=brewer.pal(8, "Dark2"))
```