Project 3

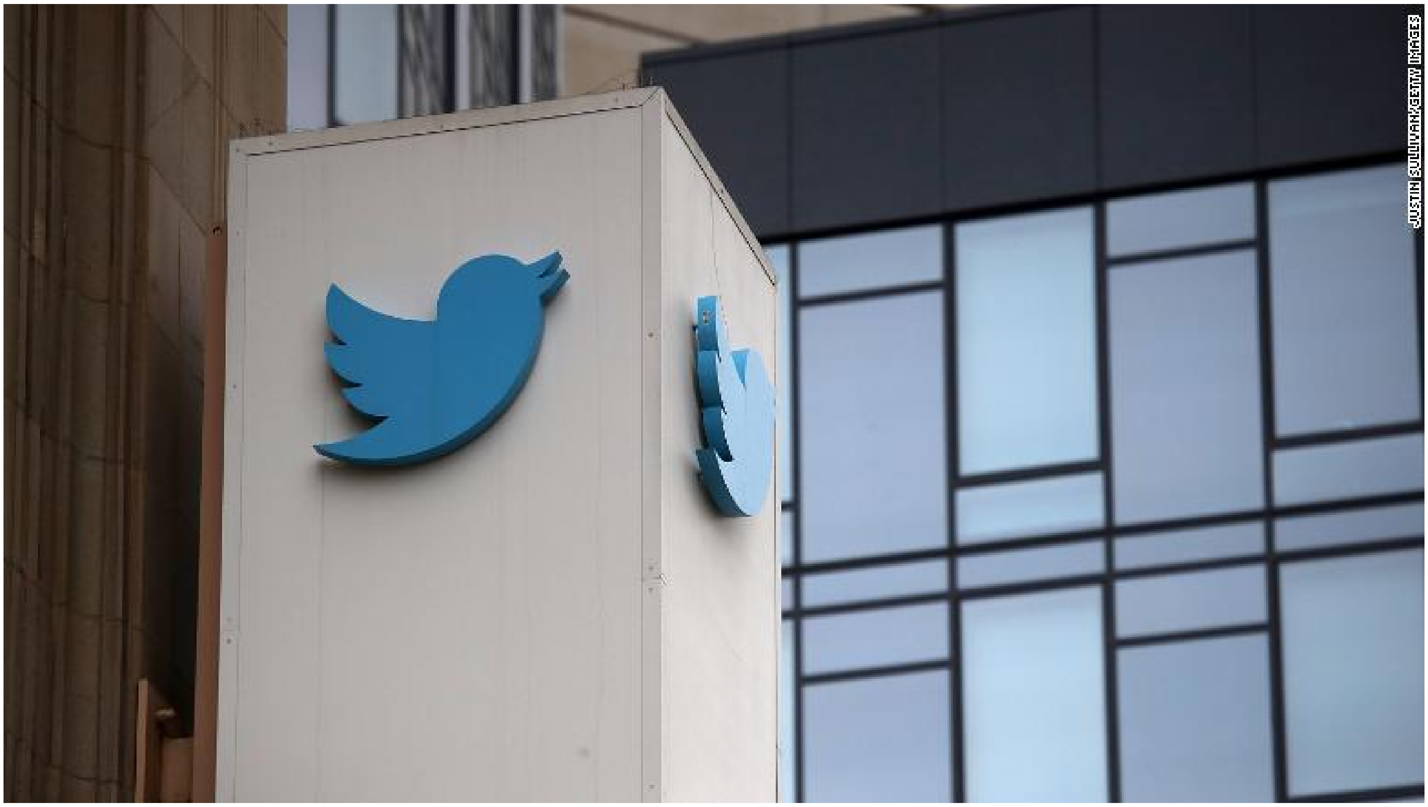
Bar Raisers - Large Group Justified

10/16/2020

DATA SCIENCE SKILLS THAT MATTER

Our group embarked on the quest to find an answer to the title question. Our approach consisted of trying to identify the terms that are commonly tagged along with the Twitter handle #Datascience. This provides the keywords that are most often associated with 'data science'. The top 20 de-duped keywords will serve as a dictionary for the analysis. It is important to note that, these keywords may not be used in professional job listings. To validate the findings from Twitter, professional job listing sites in the US such as LinkedIn and Indeed will be used.

Twitter



Twitter headquarters

Twitter is a social network founded in 2006, it has over 325 million members and serves as a barometer of public opinion. Social media has played a fundamental role in the social activism of the 21st century. Nevertheless, people share their opinions and connect in a wide range of areas including careers. Is precisely this fact that motivated us to treat Twitter as a barometer for what hashtags people associate most commonly to Data Science.

Getting Twitter Data

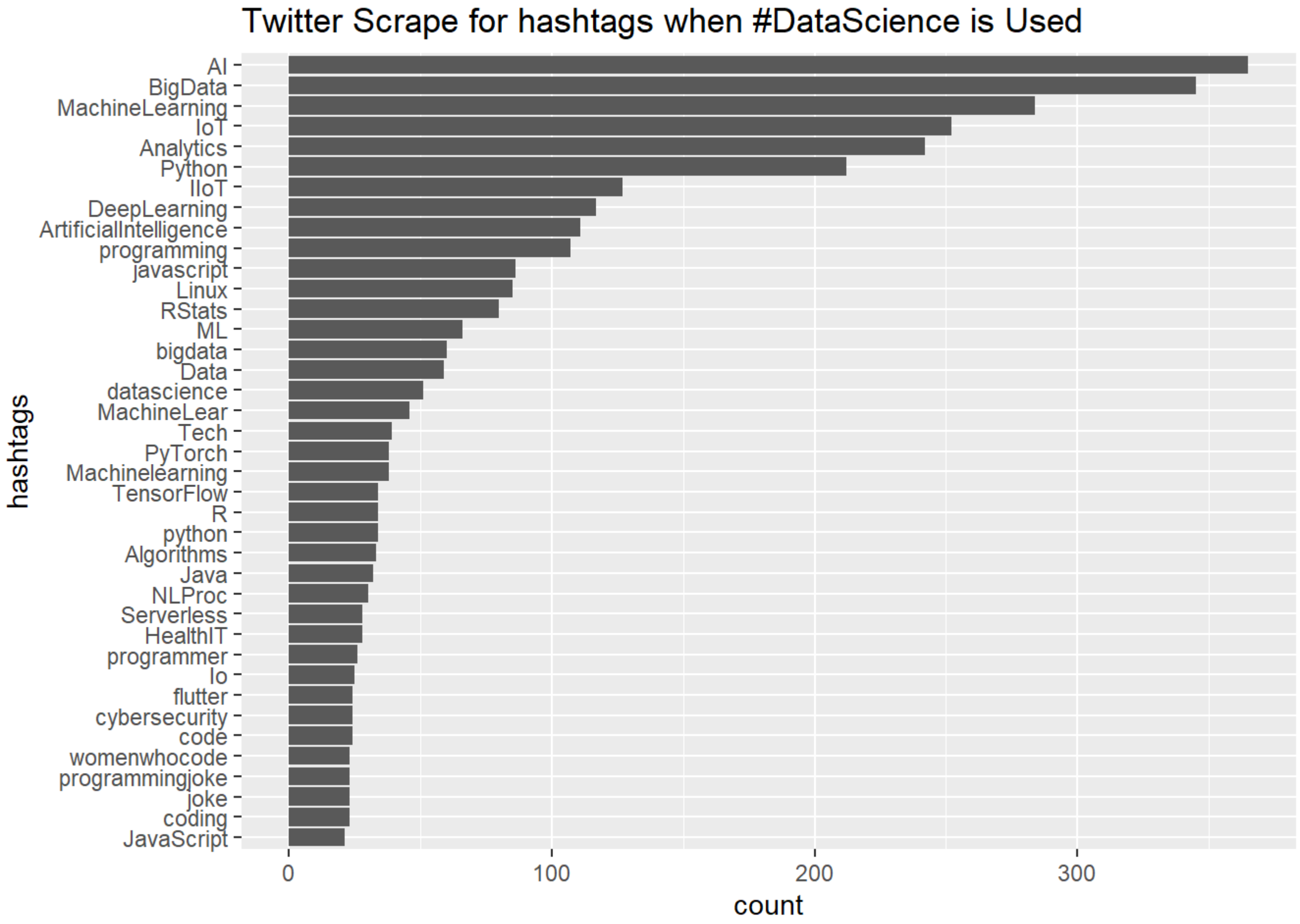
To have access to the Twitter data, we used the twitteR package. An API account and app had to be created to be able to access Twitter’s API. After connecting to twitter, we asked for the top 1000 tweets containing the hashtag #DataScience. We extracted the words from these tweets and got 5200 words. If we group and count the words that are repeated we get about 493 words.

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

word	word_count
AI	365
BigData	345
MachineLearning	284
IoT	252
Analytics	242
Python	212

```
count_word_cut<-count_word%>%
  filter(word_count>20)

ggplot(count_word_cut, mapping=aes(x=reorder(word,word_count),y=word_count))+
  geom_bar(stat="identity")+
  coord_flip() +
  labs(title="Twitter Scrape for hashtags when #DataScience is Used",x="hashtags", y="count")
```



As we can see in this plot, the terms most people associate with Data Science are: - AI, Big Data, Machine Learning and Analytics

We see different languages associated with Data Science, of which *Python* appears to be number one.

We also see frameworks like Tensorflow and PyTorch.

LinkedIn

LinkedIn

Started in 2003, LinkedIn began as a social network for professionals and has evolved throughout the years into a Networking platform with career building capabilities, training and job search functionality. LinkedIn pivoted from a social network to a full fledged enterprise tailored to career searching, training and networking. Linkedin has topped at 315 million members and, according to recent statistics, the number of business professionals in the world is estimated between 350 and 600 million individuals. This means that over 50% of the business professionals in the planet are on LinkedIn!* See source (https://thelinkedinman.com/history-linkedin/#:~:text=LinkedIn%20started%20out%20in%20the,the%20New%20York%20stock%20exchange.)

Being able to query the job listings in LinkedIn would be an invaluable resource in answering the proposed question. Scraping data from LinkedIn used to be easier a few years ago but LinkedIn has implemented some security measures to ensure its members data is kept safe. At the same time, they have made scraping job postings harder. Nevertheless, scrape we shall.

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

Supporting Scripts

As part of this project we wrote several supporting scripts and functions that helped us organize, scrape, and make the overall project more efficient. Here we list some of them but you can find them all within the appropriate folders:

- *functions*. Holds common functionality
- *scraping*. Holds the scraping functions

AWS Connection Script

```
library(DBI)
library(tidyverse)

getDbConnection <- function() {
  # This function returns a DB Connection to AWS
  password_file<-"C:\\password-files-for-r\\AWS_login.csv"

  # read in login credentials
  df_login<-read.csv(password_file)

  # Uncomment this next line to see the values
  # cat("Host: ", vardb_host, " Schema=", vardb_schema, " username=", vardb_user, " password=",
 vardb_password)
  mydb = dbConnect(RMySQL::MySQL(),
                   user=df_login$login_name,
                   password=df_login$login_password,
                   port=3306,
                   dbname=df_login$login_schema,
                   host=df_login$login_host)

  return(mydb)
}
```