Week 6. Big Data Analytics
Text mining from the web

Hyeonsu B. Kang
hyk149@eng.ucsd.edu

April 2016

1 Retrieving tweets from Twitter using Twitter Developer API

You need to activate your phone number with your Twitter account. Please visit Twitter website and create an account if you do not have one, and activate your mobile phone associated with it under Settings.

Once you have the account, please visit `https://apps.twitter.com/` and create an application as follows:

Figure 1: Twitter application creation 1

Click the "Create New App" button. You will be advanced to the following screen.

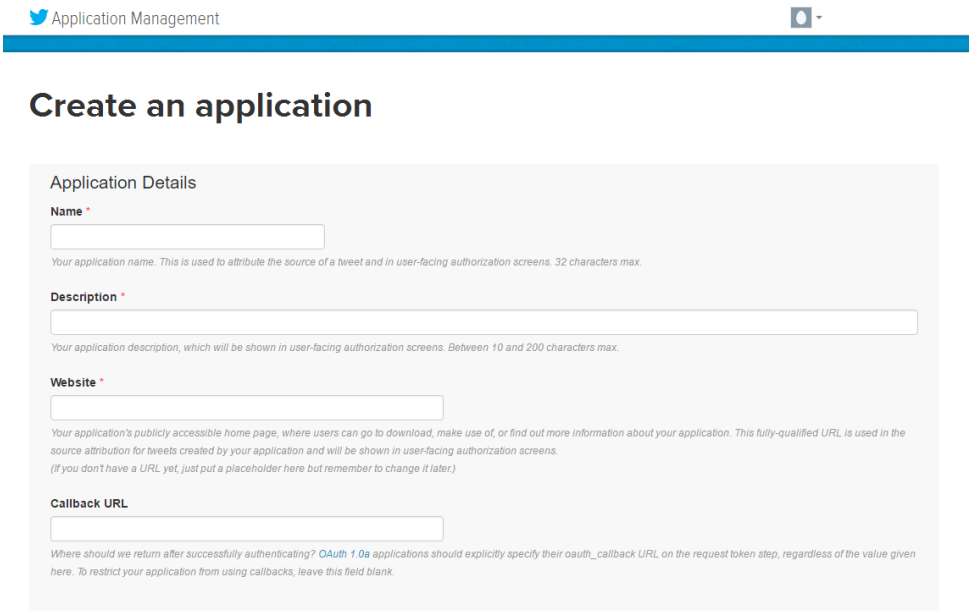

Figure 2: Twitter application creation 2

Please put in the name of the application (it could be anything that is not already taken.) For the URL, unless you are planning to publish your application on the internet later, it can be anything with preceding `http://`.

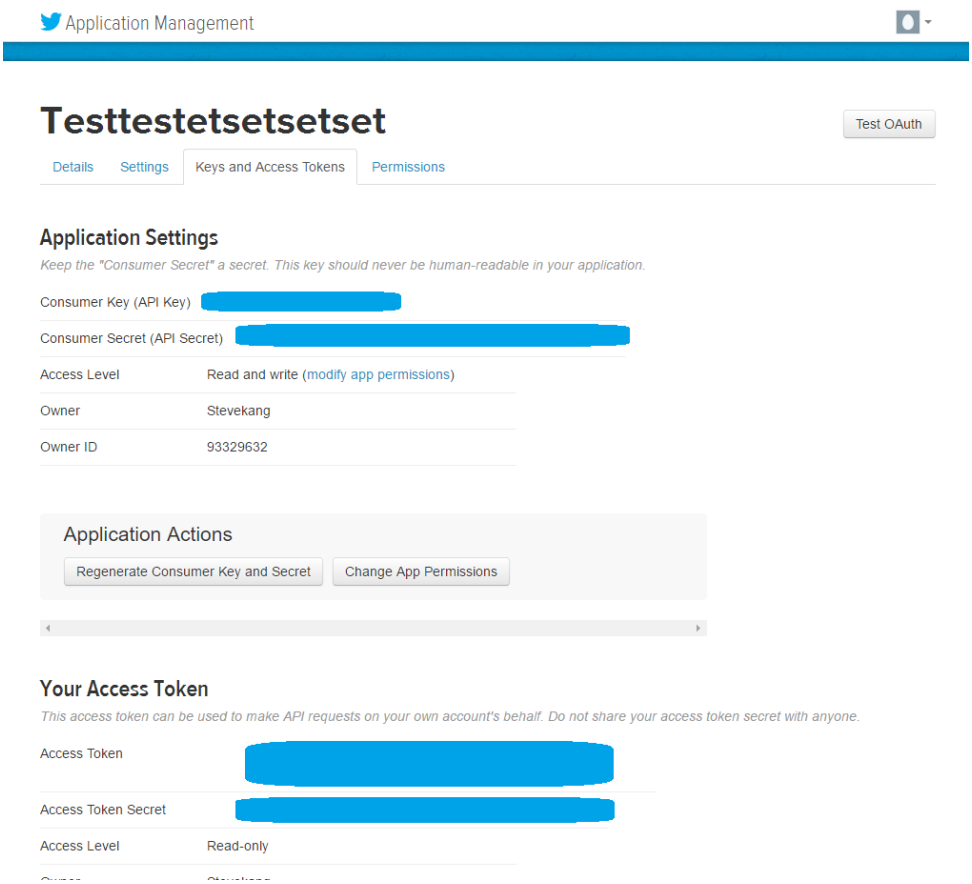

Figure 3: Twitter application creation 3

Once you successfully created an application, you can access your Consumer Key, Consumer Secret, Access Token and Access Token Secret as above. Please copy the content and save them as strings as follows.

```
> api_key = "XXXXXXXXX" # your api_key
> api_secret = "XXXXXXXXXX" # your api_secret
> access_token = "XXXXXXXXXX" # your access_token
> access_token_secret = "XXXXXXXXXX" # your access_token_sceret
> setup_twitter_oauth(api_key,api_secret,access_token,
+                     access_token_secret)
> tweets = searchTwitter("#nba", n=100)
> # what is the difference?
> tweets = searchTwitter("#nba", n=100, lang="en")
```

Reference: Jeff Gentry, *Twitter client for R*, `http://geoffjentry.hexdump.org/twitteR.pdf`. Many of the internet platforms provide functionality for free through their own Application Programming Interface (API). Facebook Analytics `https://developers.facebook.com/products/analytics`, Google API `https://console.developers.google.com/apis/library?project=omega-cider-108420&pli=1` and many more!

2 Turning a PDF file into a corpus

A simple way to get a corpus from a pdf file full of text is to read-in a pdf as a data frame and then using `VectorSource()`. We first need a program called `pdftotext`. Download it from here `http://www.foolabs.com/xpdf/download.html` and unzip it. Now to use this program in R, type the commands below:

```
> dest = "./" # the current directory
> files = list.files(path = dest, pattern = "pdf", full.names = TRUE) # list PDFs
> sapply(files, function(i) { # get rid of blanks in file names
+     file.rename(from = i, to = paste0(dirname(i), "/", gsub(" ", "", basename(i))))
+ })
./14-7955_aplc.pdf
             TRUE
> pdftotext = "C:/Users/Hyeonsu/xpdfbin-win-3.04/bin64/pdftotext.exe" # change this
> filesWithoutSpace = list.files(path = dest, pattern = "pdf", full.names = TRUE)
```

```
> lapply(filesWithoutSpace, function(i) { # use the pdftotext program to extract
+     system(paste(pdftotext, paste0('"', i, '"')), wait = FALSE)
+ })
[[1]]
[1] 0
```

0 means normal exit (without errors). Once you are done, you will have a text file extracted in your project's working directory (Q: how to find where your R project's working directory is? `getwd()`). Now, load the extracted text file and create a corpus out of it as follows

```
> textfiles = list.files(path = dest, pattern = "txt", full.names = TRUE)
> library(tm)
> corpus = Corpus(DirSource(dest, pattern = "txt"))
> stemmedCorpus = tm_map(corpus, stemDocument)
> stemmedCorpus[[1]]$content
```

Of course, we will need more processing before or after `tm_map(corpus, stemDocument)` for analysis.

3 Building a `tf-idf` matrix

Download the data files (`yelpReview1, yelpReview2, yelpReview3`) from the course website. Create a directory named `data` in your project folder and place the data files in it.

```
> data = Corpus(DirSource("data"))
> corpus  <-Corpus(DirSource("data"), readerControl = list(blank.lines.skip=TRUE))
> corpus <- tm_map(corpus, removeWords, stopwords("english"))
> corpus <- tm_map(corpus, stripWhitespace)
> corpus <- tm_map(corpus, stemDocument, language="english")
> terms <-DocumentTermMatrix(corpus,control = list(weighting = weightTfIdf))
> apply(terms, 1, function(x) {
+     x2 <- sort(x, TRUE)
+     x2[x2 >= x2[3]]
+ })
$yelpReview1.txt
       and      brief    connect    employe        far
 0.0633985  0.0633985  0.0633985  0.0633985  0.0633985
    flight    flight.      grace hopefully,       kind
 0.0633985  0.0633985  0.0633985  0.0633985  0.0633985
     layov      littl   luckily,       make       next
 0.0633985  0.0633985  0.0633985  0.0633985  0.0633985
   phoenix   phoenix.   pleasant    presenc      thank
 0.0633985  0.0633985  0.0633985  0.0633985  0.0633985
      time      time.     travel
 0.0633985  0.0633985  0.0633985

$yelpReview2.txt
     belli        bun       care      craft     drink.
0.08341908 0.08341908 0.08341908 0.08341908 0.08341908
   everyth     featur      menu.       much      nobuo
0.08341908 0.08341908 0.08341908 0.08341908 0.08341908
      pork       show      start     stout.     talent
0.08341908 0.08341908 0.08341908 0.08341908 0.08341908
      then      uniqu
0.08341908 0.08341908

$yelpReview3.txt
     cooki        ice      cream     cream.        the
0.14859023 0.14859023 0.09906016 0.09906016 0.09906016
```