Extracting Tweets with R

Twitter is a popular microblogging site.

Tweets are up to 140 characters.

Companies, organizations and individuals have found a lot of ways to utilise this data and extract meaningful information from it.

Examples include tracking earthquakes, predicting stock market, understanding relationship among users, recommending books, sentiment analysis for a product etc.

Create a Twitter application

Extract Tweets

Fig1: Extracting Tweets with R

**Steps to Create Twitter Application**

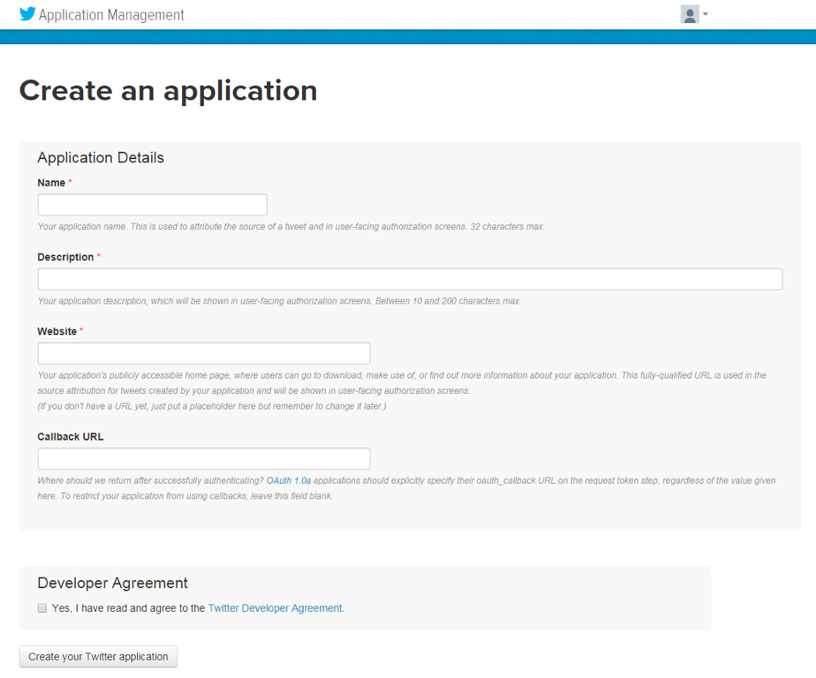
1. Use your Twitter credentials to login <https://developer.twitter.com/en/apps> and click on ‘Create New App’

2. Fill in the application form (shown below) with relevant details.

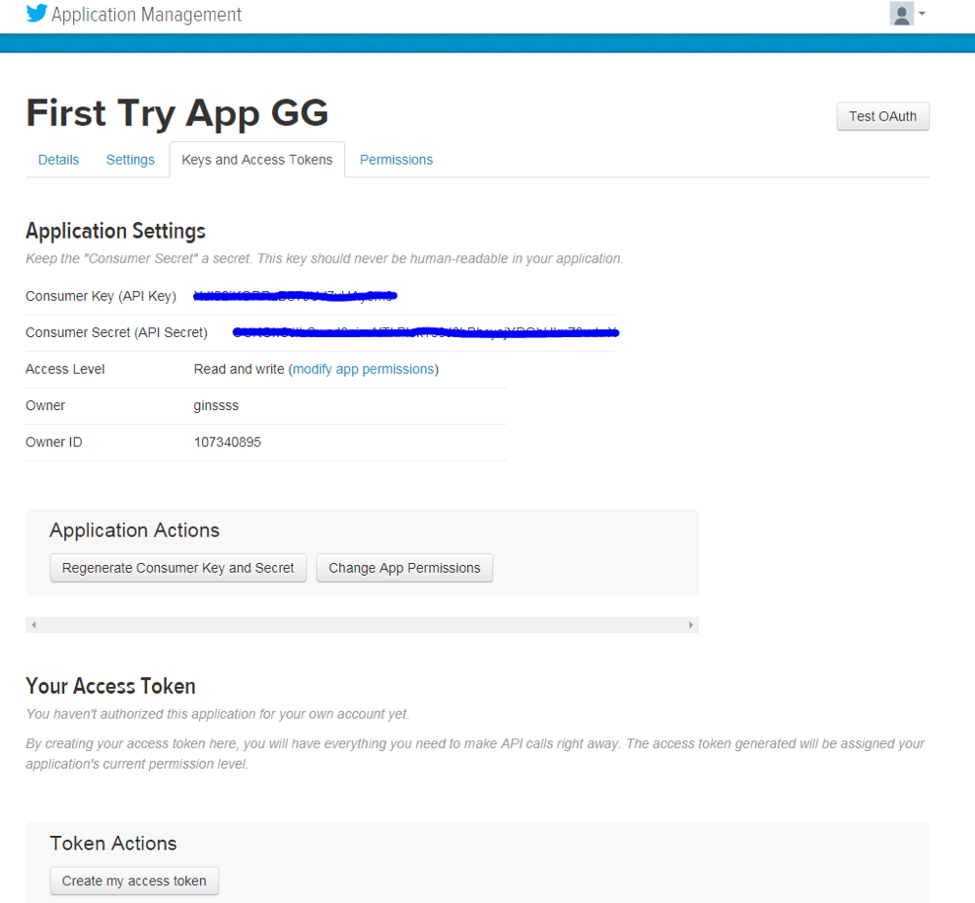
Note that the name should be unique and should not have been used by anyone else before.

After you have read the Twitter Developer Agreement, tick the check box and ‘Create your Twitter application’.

And, you will need to have your phone linked to twitter to create an app.



3. You will lend on application details page; move to ‘Keys and Access Tokens’ tab, scroll down and click ‘Create my access token’. Note the values of API Key and API Secret for future use. Thou shan’t share these with anyone, one can access your account if they get the keys.



4. In order to extract tweets, you will need to establish a secure connection between R and Twitter as follows:

Load necessary R packages and get CURL certification. ROAuth: R interface for OAuth, the open standard for token-based authorisation on the internet.

#Clear R Environment

rm(list=ls())

#Load required libraries

install.packages("twitteR")

install.packages("ROAuth")

library("twitteR")

library("ROAuth")

# Download the file and store in your working directory

download.file(url= "http://curl.haxx.se/ca/cacert.pem", destfile= "cacert.pem")

Set the certification at Twitter by making a call to OAuthFactory function.

#Insert your consumerKey and consumerSecret below

credentials <- OAuthFactory$new(consumerKey='XXXXXXXXXXXXXXXXXX',

consumerSecret='XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX',

requestURL='https://api.twitter.com/oauth/request\_token',

accessURL='https://api.twitter.com/oauth/access\_token',

authURL='https://api.twitter.com/oauth/authorize')

Let’s now ask Twitter for access!

credentials$handshake(cainfo="cacert.pem")

5. After executing the above code, you will be directed to Twitter’s authorisation screen. Click on Authorize App and note the PIN generated. Go back to RStudio and enter the PIN. Note, you will only need to do this once.

Save the credentials for later use:

save(credentials, file=”twitter authentication.Rdata”)

**Extract Tweets**

Now that we are all done with setting up gateways to reach Twitter, let’s get our hands dirty with real data. Function searchTwitter lets you search through Twitter and return a list of tweets consisting the searched text.

Below is a piece of code to extract tweets with the search string, #DataLove. Explore other parameters of this function that lets you filter for time period, geography etc.

#Load Authentication Data

load(“twitter authentication.Rdata”)

#Register Twitter Authentication

setup\_twitter\_oauth(credentials$consumerKey, credentials$consumerSecret, credentials$oauthKey, credentials$oauthSecret)

#Extract Tweets with concerned string(first argument), followed by number of tweets (n) and language (lang)

tweets <- searchTwitter('#DataLove', n=10, lang="en")

#A sample code for extraction of tweets, cleaning and doing lexicon analysis.

library(twitteR)

library(RCurl)

library(textclean)

library(tm)

library(textclean)

library(lexicon)

consumer\_key<-'tUXKJclac8xZzE7N0177ntNOS'

consumer\_secret<-'GPsbxm9gS8vry2Y8JuMfzmC7hf6doc5IWcsuU6K2L8SuhBAvOH'

access\_token<-'1105007311297363969-eUaSv2WuTgE1YGg36C0OIRpeN4nhX9'

access\_secret<-'jHZTDKoHfIjlDm4oou8bv2mf5FnzcQyCaCqQGdabKlg1I'

setup\_twitter\_oauth(consumer\_key,consumer\_secret,access\_token,access\_secret)

tweet11<-searchTwitter("DREAM11",n=150,lang ='en')

tweet11

df<-do.call("rbind",lapply(tweet11,as.data.frame))

df

df["DuplicateFlag"]=duplicated(df$text) #to remove retweeted tweets

df$text<-replace\_internet\_slang(df$text) #to expand abbreviations

df$text<-replace\_contraction(df$text) #to expand contractions

df$text<-replace\_emoji(df$text) # to replace emoji with their name

df$text<-removeWords(df$text,stopwords()) #to remove stopwords

df$text<-gsub(pattern="\\d",replace=" ",df$text) #to remove digits

df$text<-gsub("[[:punct:]]", " ", df$text) #to remove punctuations

df$text<-gsub("(f|ht)+p(s?)://(.\*)[.][a-z]+"," ",df$text) #to remove hyperlinks

df$text<-gsub(pattern="\\b[a-z]\\b{1}","",df$text) #to remove occurence of single word

df$text<-stripWhitespace(df$text) ##to strip white space

df$text<-tolower(df$text) ##to convert it in lower case

write.table(df$text, file = "ss.txt", sep = ",", quote = FALSE, row.names = F) ##to save a file in text file

library(sentimentr)

sentiment(df$text)

extract\_sentiment\_terms(df$text)

plot(sentiment(df$text))