Quantitative Text Analysis
Exercise 10: Data Mining from twitter

1st August 2014, Essex Summer School

Kenneth Benoit and Paul Nulty

In this exercise you will try out R code for using the Twitter REST and streaming APIs, and put the results into a `quanteda` corpus for analysis.

Open RStudio and install and load the twitteR and streamR packages:

```
library(devtools)
install_github("twitteR", username="geoffjentry")
install_github("streamR", "pablobarbera", subdir="streamR")
library(twitteR)
library(streamR)
library(quanteda)
```

Instructions

1. Extracting Twitter data: The REST API

   (a) The authentication for the REST API uses the four keys that you got after completing the application form on the twitter developers page. The function `setup_twitter_oauth` in the twitter api will connect your R application using these:

   ```
   setup_twitter_oauth(consumer_key = '',
                       consumer_secret ='',
                       access_token='',
                       access_secret='')
   ```

   (b) Look at the documentation: `help(package="twitteR")` and make a simple search.

   ```
   results <- searchTwitter('juncker', n=50)
   #transform the results object into a data frame for inspection
   df <- as.data.frame(t(sapply(results, as.data.frame)))
   ```

   (c) Look up information about one of the users from the screen names in the results dataframe.

   ```
   #get information about a user
   user <- getUser(df$screenName[40])
   usdf <- as.data.frame(user)
   ```

2. Applying `quanteda` functions to results.

   (a) There is a `quanteda` function to package the search command and create a corpus, collect a corpus of 1000 tweets mentioning UK opposition leaders with this command:

```
twitCorp <- twitterTerms("miliband OR farage", numResults=500,
                    'consumerkey',
                    'consumersecret',
                    'accesstoken',
                    'accessecret')
twitCorp$attribs$texts <- iconv(twitCorp$attribs$texts , from="latin1", to="UTF-8")
```

(b) Make a dfm from the corpus, after removing retweets: remember you can subset from a corpus object if you want to remove certain categories of tweets:

```
twitCorp <- subset(twitCorp)
```

(c) Run a dictionary analysis using the Laver and Garry dictionary on the results (or populism if you prefer) — the process will be the same as in exercise five (instructions and solution on website).

3. (OPTIONAL!) Using streamR

(a) Full instructions for authenticating, searching, and mapping using the streaming API are available on the developer's website: https://github.com/pablobarbera/streamR. (Pablo Barbera).