Twitter Crawling using R

Accessing Twitter using R

Siti Mariyah

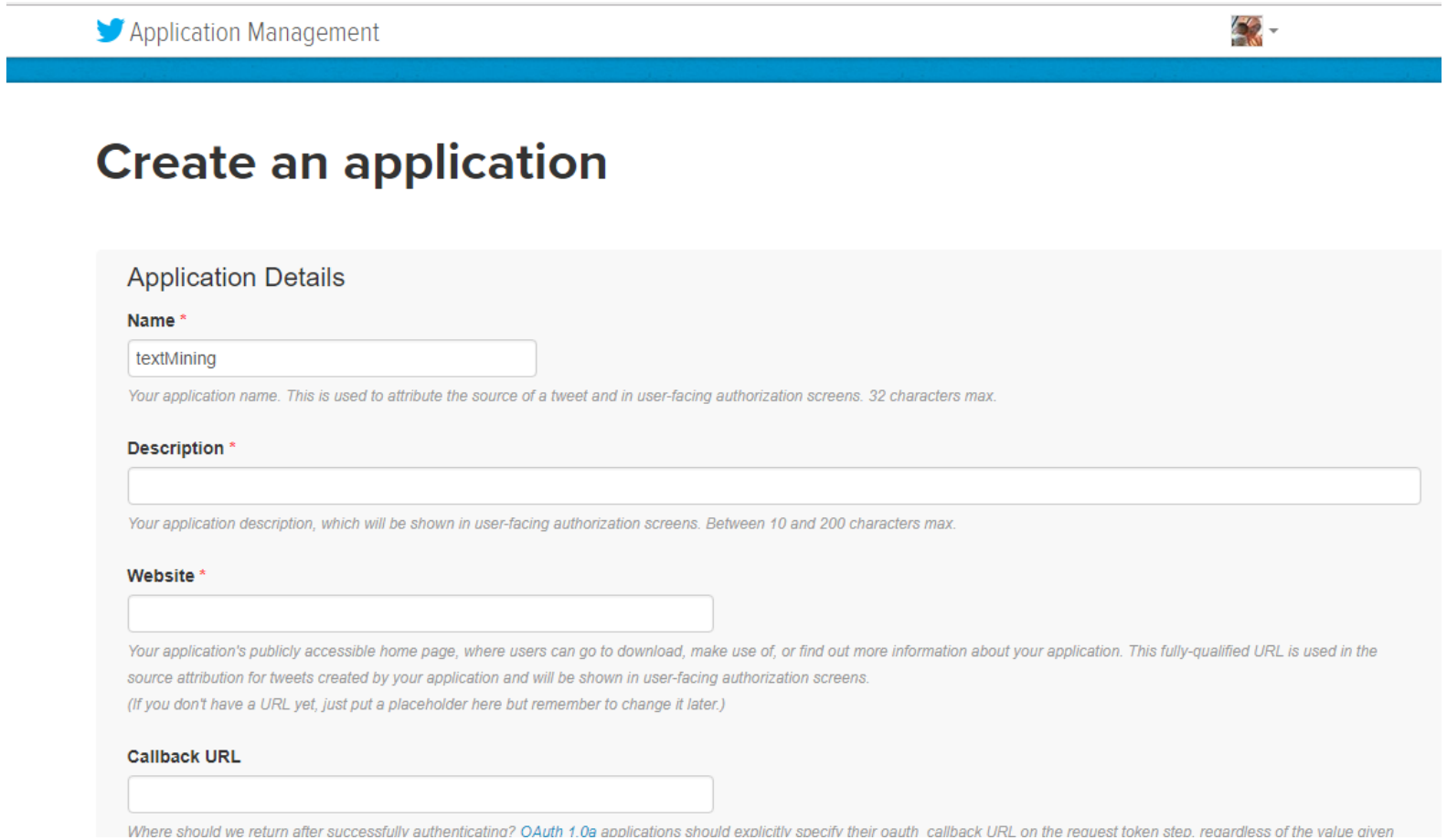
Tahapan

1. Download Package twitter

   ```
   install.packages("twitterR")
   ```

2. Buat Twitter Account
3. Sign in Twitter Account
4. Kunjungi link https://apps.twitter.com/
5. Buat aplikasi baru melalui Application Management melalui link ini https://apps.twitter.com/app/new
6. Lengkapi data yang diperlukan seperti nama aplikasi, deskripsi aplikasi, url website, dan callback URL

Tahapan


Application Management
Create an application
Application Details
Name *
textMining
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.
Description *
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.
Website *
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)
Callback URL
Where should we return after successfully authenticating? OAuth 1.0a applications should explicitly specify their oauth_callback URL on the request token step, regardless of the value given

Tahapan

7. Setelah data lengkap diisi lalu klik “Create Your Twitter Application”
8. Jika berhasil maka Anda akan mendapatkan informasi Applcation Settings seperti access level, consumer key (API key), Callback URL, Callback URL Locked, Sing in with Twitter, App-only Authentication, Request Token URL, Authorize URL, Access Token URL
9. Panggil library twitter

```
library(twitterR)
```

Application Management

text mining solusi247

Test OAuth

Details | Settings | Keys and Access Tokens | Permissions

Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	pJ8fyRKpxDjWX1DKErDnWzkZc
Consumer Secret (API Secret)	ngTsIRCNgMD50TqFGp8Vhtw2QEorFW3B1XIuW3IloLWxZgv1F7
Access Level	Read and write (modify app permissions)
Owner	nada_mommy
Owner ID	2896928972

Application Actions

Regenerate Consumer Key and Secret | Change App Permissions

Tahapan

10. Masukkan informasi consumer key, consumer secret, access token dan access secret ke dalam variabel

```
consumer_key <- “your consumer key”
consumer_secret <- “your consumer secret”
access_token <- “your access token”
access_secret <- “your access secret”
```

11. Setting Twitter Authentication

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

Tahapan

12. Searching Tweets berdasarkan keyword tertentu, missal keywordnya adalah “bca” dengan jumlah tweets sebanyak 5000

```
searchTwitter(‘bps statistics’, n=5000)
```

Ini adalah contoh sederhana dari crawling tweets. Secara umum searchTwitter menyediakan:

Format is searchTwitter("Search Terms", n=100, lang="en", geocode="lat,lng", also accepts since and until).

```
searchTwitter(“bca OR kartu kredit OR ‘Bank Central Asia’ OR bank”, n=10, lang=“id”, geocode=“6.1751, 106.8650”, since="2017-01-01")
```

Format Contoh Penulisan Keyword

Operator	Behavior
Obamacare ACA	will find tweets containing both "Obamacare" and "ACA"; not case sensitive
Obamacare ACA	will find tweets containing the exact phrase "Obamacare ACA"; not case sensitive
Obamacare OR ACA	will find tweets containing either "Obamacare" or "ACA" or both; not case sensitive; the OR operator IS case sensitive.
Obamacare -ACA	will find tweets containing "Obamacare" but not "ACA
#Obamacare	will find tweets containing the hashtag "Obamacare"
from:BarackObama	will find tweets sent from Barack Obama
to:BarackObama	will find tweets sent to Barack Obama
@BarackObama	will find tweets referencing Barack Obama's account
Obamacare since:2014-08-25	will find tweets containing "Obamacare" and sent since 2010-08-25 (year-month-day)
ACA until:2014-08-22	will find tweets containing "ACA" and sent before 2010-08-25

There are a few other query operators that you can review on the Twitter Search

Tahapan

13. Hasil crawling disimpan dalam variabel

```
bpsTweets <- searchTwitter(“bps OR badan pusat statistik OR ‘bps ri’, n=10, lang=“id”, since="2017-01-01")
typeof(bpsTweets)
```

14. Hasil crawling disimpan ke dalam file txt atau csv

```
capture.output(bpsTweets, file=“bpsTweets.txt”)
capture.output(bpsTweets, file=“bpsTweets.csv”)
```

Tahapan

15. Transformasi list tweets ke dalam sebuah data frame

```
tweetsDF <- twListToDF(bpstweets)
```

16. Tampilkan summary dari tweetsDF. Summary tweets menyajikan informasi jumlah observasi, jumlah variabel, dan value dari masing-masing variabel. Variabelnya seperti text (tweets sebenarnya), created (timestamp ditulisnya tweet tersebut), ID (id unik dari tweet), longitude dan latitude.

Mengelola Text Tweets

1. Contoh dengan satu kalimat saja. Ambil satu tweet text.

```
textTweets <- tweetsDF$text
typeof(textTweets)
text1 <- textTweets[1]
```

2. Text1 : "@HaloBCA Min.. tanya dong, ane mau buka rek BCA, bisa ga yah.. didaerah Jakarta, tapi KTP ane Alamat Bekasi...“
3. Packages yang dipakai:

```
library(tm)
library(stringr)
```

Mengelola Text Tweets

4. Mengubah ke lowercase semua

```
lower_text1 <- tolower(text1)
```

5. Menambahkan suatu string ke string yang sudah ada

```
text2 <- “ini string tambahan”
text1 <- paste(text1, text2, sep= “ ”)
```

6. Mencari keberadaan suatu kata/string dalam suatu kalimat. Misal mencari keberadaan string “bps”

```
str_extract(text1, “bps”)
str_extract_all(text1, “bps”)
str_extract_all(text1, “bps”)
```

Mengelola Text Tweets

- Pencocokkan karakter sifatnya case sensitive, huruf capital berbeda dengan huruf kecil, “learn” != “Learn”

7. Transformasi text twitter

```
toSpace <- content_transformer(function(x, pattern) gsub(pattern, “ ”, x))
text1 <- tm_map(text1, toSpace, “/”)
text1 <- tm_map(text1, toSpace, “@”)
text1 <- tm_map(text1, “\\|”)
```

Mengelola Text Tweets

8. Membuang Angka

```
text1 <- tm_map(text1, removeNumbers)
text1
```

9. Membuang tanda baca (punctuation)

```
text1 <- tm_map(text1, removePunctuation)
```

10. Membuang stopwords

```
text1 <- tm_map(text1, removeWords, stopwords("english"))
length(stopwords("english"))
stopwords("english")
```

Mengelola Text Tweets

11. Membuang Stop Words Sendiri

```
text1 <- tm_map(text1, removeWords, c("department", "email"))
```

12. Membuang Strip Whitespace

```
text1 <- tm_map(text1, stripWhitespace)
```

13. Melakukan Stemming

    Terlebih dulu install packages SnowballC

```
install.packages("SnowballC")
library(SnowballC)
text1 <- tm_map(text1, stemDocument)
```

14. Membuat *Document Term Matrix*

```
dtm <- DocumentTermMatrix(text1)
```

Mengelola Text Tweets

15. Inspect hasil *document term matrix*

```
class(dtm)
dim(dtm)
inspect(dtm[, 10:20])
```

16. Membuat*Term Document Matrix*

```
tdm <- TermDocumentMatrix(text1)
```

17. Eksplorasi *Document Term Matrix*

```
freq <- colSums(as.matrix(dtm))
length(freq)
ord <- order(freq)
freq[head(ord)]
freq[tail(ord)]
```

Mengelola Text Tweets

18. Membuat wordcloud dari tweets yang sudah dibersihkan tadi. Karena kita sudah menghitung frekuensi kata-kata (variabel freq) maka kita tidak perlu lagi membuat wordcloud dari awal.

```
install.packages("wordcloud")
install.packages("RColorBrewer")
library(wordcloud)
library(RColorBrewer)
set.seed(123)
wordcloud(names(freq), freq, min.freq=40)
```