

# Twitter client for R

Jeff Gentry

May 19, 2010

## 1 IMPORTANT NOTICE

The old Twitter authentication mechanism is being removed in June, 2010 in favor of OAuth. As of this writing there is not an OAuth solution for R, so unless someone (or myself) writes such a beast you will not be able to access any authenticated aspects of Twitter.

## 2 Introduction

Twitter is a popular service that allows users to broadcast short messages ('*tweets*') for others to read. These can be used to communicate with friends, to display headlines, for restaurants to list daily specials, and more.

The *twitteR* package is intended to provide access to the Twitter API within R. Using the S4 classes *user*, *status* and *directMessage*, users can read and write information via the Twitter server.

### 2.1 Disclaimer

At this time, the package does not yet fully support the Twitter API. Currently basic operations are supported but there are still gaps in searching capabilities as well as account administration. Also, many commands do not yet support all possible arguments to calls. Lastly, because many authenticated action would require a valid Twitter login, various operations in this vignette are pregenerated.

## 3 Getting Started

The first basic to know is the `initSession` command. This will provide you with a token that you will be passing around to most/all of the other calls from the *twitteR* package. Any functionality which does not require you to be authenticated will work without a session token, but will not work otherwise.

```
> library(twitteR)
```

To use the `initSession` command, supply your username and password, and assign this to a variable:

```
> sess <- initSession('myname', 'mypassword')
```

## 4 Time to talk about timelines

Just as there are various *timelines* in Twitter, the *twitteR* package provides various interfaces to access them. The first and most obvious would be the *public timeline*, which retrieves the 20 most recent public tweets on Twitter. As this is fully public, we do not need to use our session token, although it won't hurt to do so.

```
> publicTweets <- publicTimeline()
> length(publicTweets)

[1] 20

> publicTweets[1:5]

[[1]]
[1] "rucamend: Tem um tempÃo que nÃo assistia MalhaÃo, assisti ontem e hoje e ainda con

[[2]]
[1] "SapulNews5: INQUIRER: BDA ITONG SI KOALA BEAR.SIYA SI ALYS ROBIN, NA STARRNG ROLE SA IM

[[3]]
[1] "young_moneey_: Cant touch this. tananana nana nana Cant touch this. ;)"

[[4]]
[1] "FernandoSLeitao: ... entre um e outro, nos livra do sectarismo, combustÃvel dos confli

[[5]]
[1] "RadioWong: Guitarists- is there ANYBODY in the world who uses the thumb-over method for
```

Similarly, we can look at a particular user's timeline. In this case, whether or not you need to include your session token depends on if their feed is protected or not. For this example, let's use a public user, *cranatic*. Note that you can specify either a user's screen name, or an actual *user* object to the function.

```
> cranTweets <- userTimeline("cranatic")
> cranTweets[1:5]

[[1]]
[1] "cranatic: Update: GSM, JOP, Matrix. http://crantastic.org/daily/20100519 #rstats"

[[2]]
```

```
[1] "cranatic: New: DAMisc. http://crantastic.org/daily/20100519 #rstats"
```

```
[[3]]
```

```
[1] "cranatic: Update: BayesQTLBIC, MplusAutomation. http://crantastic.org/daily/20100518 #rstats"
```

```
[[4]]
```

```
[1] "cranatic: Update: EbayesThresh, PTAk. http://crantastic.org/daily/20100517 #rstats"
```

```
[[5]]
```

```
[1] "cranatic: New: CollocInfer. http://crantastic.org/daily/20100517 #rstats"
```

The `userTimeline` function currently only supports the default of returning the 20 most recent tweets.

Suppose you wish to see what your friends are up to, for this we have `friendsTimeline`, which will return the 20 most recent tweets from your friends. This function does require you to use your login token:

```
> friendTweets <- friendsTimeline(sess)
```

Lastly, you can see a list of the 20 most recent tweets in which you were mentioned via `mentions`. As with `friendsTimeline`, this requires your login token

```
> myMentions <- mentions(sess)
```

## 4.1 Searching Twitter

The `searchTwitter` function can be used to search the Twitter timeline. Example searches are such things as hashtags, basic boolean logic such as AND and OR. The `num` argument can be used to specify the number of tweets to return, defaulting to 25.

```
> sea <- searchTwitter("#twitter", num = 50)
```

```
> sea[1:5]
```

```
[[1]]
```

```
[1] "PaperPlaneWayne: @IJ@iLoveShortyDip: #TeamiPhone Download App #Twitter If Not u Wack F"
```

```
[[2]]
```

```
[1] "KingMarco18: oooo apparently there is an official #Twitter for the #Iphone !"
```

```
[[3]]
```

```
[1] "cylab: Reuters: #Twitter Expects Hundreds of Advertisers This Year http://bit.ly/bemULC"
```

```
[[4]]
```

```
[1] "OhMyWhitegirl: RT @FS_WHITEBOY: This is a #Twitter & #Facebook update>>>&g"
```

```
[[5]]
```

```
[1] "ChefCoyarDee: @KoOkLa Nah we met on #Twitter but we homies now cuz we part of the same"
```

## 4.2 Seeing what other R folks are up to

The `Rtweets` function will retrieve the 20 most recent tweets that carry the `rstats` hash tag, which is commonly used by members of the *R* community. For this command, using your login token is optional - it is unlikely to change the results, but it is possible for that to happen:

```
> rt <- Rtweets()
> rt[1:5]

[[1]]
[1] "jaredlander: @drewconway How about an #rstats meetup about translating R code into cust

[[2]]
[1] "cranatic: Update: GSM, JOP, Matrix. http://crantastic.org/daily/20100519 #rstats"

[[3]]
[1] "cranatic: New: DAMisc. http://crantastic.org/daily/20100519 #rstats"

[[4]]
[1] "Chris_Bilder: RT @Rbloggers: useR-2010 is looking for a T-shirt design http://goo.gl/ft

[[5]]
[1] "revodavid: Just finished my slides for Introduction to Revolution R webinar (live tomor
```

## 5 Looking at users

To take a closer look at a Twitter user (including yourself!), run the command `getUser`. As with many functions, whether or not you need to use your login token depends on if this user profile is protected or not.

```
> crantastic <- getUser("crantastic")
> crantastic

[1] "Crantastic"
```

Furthermore, we can look at this user's friends, as well as those following them (same disclaimer regarding the login token applies):

```
> friends <- userFriends("crantastic")
> friends[[1]]

[1] "BradleyWoodside"

> followers <- userFollowers("crantastic")
> followers[1:5]
```

```
[[1]]
[1] "keithkurson"

[[2]]
[1] "rebecca_a_moses"

[[3]]
[1] "ellydisapproves"

[[4]]
[1] "klassina"

[[5]]
[1] "megcabot"
```

## 5.1 The user class

In both of the above cases, the argument can be a string noting the user's screen name, or a *user* object. Let's look more closely at the *user* class to see what is available within it.

The following is a look at the collection of available get methods for the *user* class:

```
> curUser <- friends[[1]]
> screenName(curUser)

[1] "BradleyWoodside"

> description(curUser)

[1] ""

> tweetCount(curUser)

[1] 58

> followersCount(curUser)

[1] 16

> favoritesCount(curUser)

numeric(0)

> friendsCount(curUser)

[1] 7

> name(curUser)
```

```

[1] "Fake Brad Woodside"
> protected(curUser)
[1] FALSE
> verified(curUser)
[1] FALSE
> location(curUser)
[1] "Fredericton"
> id(curUser)
[1] 114974464
> lastStatus(curUser)
[1] "Unknown: Look stupid transit riders if you cant show respect for our bus shelters I wil

```

## 6 Tweeting

Tweeting, or updating one's status, can be done via the function `tweet` (or the identical `updateStatus`). To do this, simply post your message and include your login token. The return value is the *status* object that represents your tweet.

```
> tweet <- tweet('this is a tweet', sess)
```

### 6.1 When your tweet goes awry

Suppose you realize you've made a typo, or said something you shouldn't have. For this, we have `deleteStatus`, which takes an object of class *status* and your login token. Assuming you have the authority to delete the *status* object, it will be removed.

```
> deleteStatus(tweet, sess)
```

### 6.2 The status class

The *status* class has the following methods defined:

- *text*: Retrieves the text of the tweet
- *screenName*: Screen name of the sender
- *id*: Retrieves the ID of the tweet

- `created`: Retrieves the date the tweet was created
- `replyToSN`: If this is a reply, the screen name for the reply
- `replyToSID`: If this is a reply, the message this is in reply to
- `replyToUID`: If this is a reply, the user this is in reply to
- `favorited`: If this reply is favorited
- `statusSource`: Source of the tweet

## 7 Direct messages, aka DMs

Direct Messages (DMs) can be sent with the `dmSend` command, which takes your message, the user (either a string or a *user* object), and your login token:

```
> dm <- dmSend('this is a test', 'crantastic', sess)
```

You can also see the DMs that you've sent and received:

```
> sent <- dmSent(sess)
> rcvd <- dmGet(sess)
```

These functions all return objects of class *directMessage*, which have the following methods:

- `text`: Retrieves the text of the message
- `recipient`: Retrives the *user* object of the recipient
- `recipientSN`: Retrieves the screen name of the recipient
- `recipientID`: Retrieves the ID of the recipient
- `sender`: Retrieves the user object of the sender
- `senderSN`: Retrieves the screen name of the sender
- `senderID`: Retrieves the ID of the sender
- `id`: Retrieves the ID
- `created`: Retrieves the creation date

## 8 Session Information

The version number of R and packages loaded for generating the vignette were:

```
R version 2.11.0 Patched (2010-05-01 r51890)
x86_64-unknown-linux-gnu
```

```
locale:
```

```
[1] LC_CTYPE=en_US.UTF-8      LC_NUMERIC=C
[3] LC_TIME=en_US.UTF-8      LC_COLLATE=en_US.UTF-8
[5] LC_MONETARY=C            LC_MESSAGES=en_US.UTF-8
[7] LC_PAPER=en_US.UTF-8    LC_NAME=C
[9] LC_ADDRESS=C            LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
```

```
attached base packages:
```

```
[1] tools      stats      graphics  grDevices  utils      datasets  methods
[8] base
```

```
other attached packages:
```

```
[1] twitter_0.1.6  rjson_0.1.8    RCurl_1.4-2    bitops_1.0-4.1
```