Here we go again, using a Twitter trend as blog fodder! Colin Fay  
launched an inspiring movement by sharing his R goals of 2019.

My [#RStats](https://twitter.com/hashtag/RStats?src=hash&ref_src=twsrc%5Etfw) goals for 2019:

1⃣ Becoming entirely fluent with {data.table}

2⃣ Getting at ease with {Rcpp}

What are yours?[#rdatatable](https://twitter.com/hashtag/rdatatable?src=hash&ref_src=twsrc%5Etfw) [#rcpp](https://twitter.com/hashtag/rcpp?src=hash&ref_src=twsrc%5Etfw)

— Colin Fay (@\_ColinFay) [December 29, 2018](https://twitter.com/_ColinFay/status/1078953442385772544?ref_src=twsrc%5Etfw)

It’s been quite interesting reading the objectives of other tweeps: what  
they want to learn, make, how they want to get involved in the  
community, etc. As Mike Kearney, rtweet’s maintainer, underlined, it  
is excellent reading material!

Excellent reading material–tweets about 2019 [#rstats](https://twitter.com/hashtag/rstats?src=hash&ref_src=twsrc%5Etfw) goals: <https://t.co/6wrGeqsWbm>

— Mike Kearney (@kearneymw) [December 31, 2018](https://twitter.com/kearneymw/status/1079865464241774593?ref_src=twsrc%5Etfw)

… but also blogging material! Let me fetch and tokenize these tweets to  
summarize them!

**Disclaimer: I later saw that** [**Jason Baik got the same  
idea**](https://twitter.com/jsonbaik/status/1079967070207516672) **and was  
faster than I, find the analysis**[**here**](https://github.com/jasonbaik94/rstats-2019-goals)**.**

**Collect Twitter data**

If you’re using rtweet for the first time, check out [its  
website](https://rtweet.info/) for information about use and setup and  
also refer to [Twitter API docs](https://developer.twitter.com/)  
themselves for knowing more about rate limitation and e.g. learning that  
the search endpoint won’t let you get tweets older than 6-9 days old.

tweets <- rtweet::search\_tweets("Rstats goals 2019",

include\_rts = FALSE)

I obtained 87 tweets from 85 unique users. Definitely not big data, but  
not bad!

**Tokenize tweets**

I then set out to tokenize the tweets into words using the specific  
tokenizers::tokenize\_tweets() tokenizer via the tidytext package. If  
you’re new to tidytext I’d recommend reading the book written by [its  
authors](https://www.tidytextmining.com/). *A token in natural language  
processing can be a word, line, etc. which is a totally different  
concept from a token for rtweet functions (your API credentials).*

The tweet tokenization is a “tokenization by word that preserves  
usernames, hashtags, and URLS”. So awesome, and today is the first time  
I find an occasion to use it! I also removed stopwords.

library("magrittr")

stopwords <- rcorpora::corpora("words/stopwords/en")$stopWords

tokens <- tweets %>%

dplyr::select(text) %>%

tidytext::unnest\_tokens(token, text,

token = "tweets",

drop = FALSE) %>%

dplyr::filter(!token %in% stopwords)

**Analyze tweets**

**Most mentioned topics**

I first was able to draw a figure similar to Jason Baik’s one, with the  
most common tokens. I too removed digits.

library("ggalt")

tokens %>%

dplyr::mutate(token = stringr::str\_remove\_all(token, "[^\x01-\x7F]")) %>%

dplyr::mutate(token = stringr::str\_remove\_all(token, "[[:digit:]]")) %>%

dplyr::filter(! token %in% c("", "#rstats", "goals")) %>%

dplyr::count(token, sort = TRUE) %>%

dplyr::mutate(token = reorder(token, n)) %>%

head(n = 18) %>%

ggplot() +

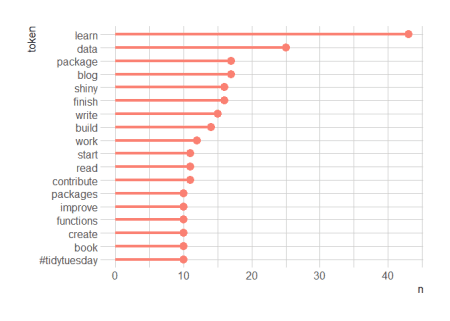
geom\_lollipop(aes(token, n),

size = 1.5, col = "salmon") +

hrbrthemes::theme\_ipsum(base\_size = 12,

axis\_title\_size = 12) +

coord\_flip()



**What actions?**

In this figure I identify verbs like *learn*, *finish*, *write*, *build*  
and *contribute*. Let me look at a sample of lines for each of them.  
This is a sample of lines for a small sample of verbs.

lines <- tweets %>%

dplyr::select(text) %>%

tidytext::unnest\_tokens(line, text,

token = "lines")

sample\_verb <- function(verb, lines){

set.seed(42)

dplyr::filter(lines, stringr::str\_detect(line, paste0(verb, " "))) %>%

dplyr::sample\_n(3)

}

samples <- purrr::map\_df(c("learn", "finish", "write", "build", "contribute"), sample\_verb, lines)

knitr::kable(samples)

| **line** |
| --- |
| 3⃣ learn how to make r packages and write my code so it could be made into an r package more easily |
| 1. learn how to do spatial analysis in r |
| 2⃣ learn better way to automate feature engineering (neural nets) for text |
| 1⃣ finally finish all the courses and certifications i started last year on #coursera and #datacamp |
| 2⃣ finish my track and field r package |
| 3). finish that text mining project i started in october |
| 4⃣ write an advanced shiny book with bookdown 🎈 |
| 3⃣ learn how to make r packages and write my code so it could be made into an r package more easily |
| 1⃣ write the htmlwidgets book |
| – build my first #rstats package (aiming for 2 but 1 would be great :d) |
| 2⃣ build a shiny web app to explore tx staar data |
| – use f(x) regularly & build own package. cease patching. |
| 5⃣ contribute to foss <https://t.co/oh7mwcq50r> |
| 2 contribute more to #rstats community through #scicomm, #stackoverflow, etc |
| 3) contribute to #swdchallenge (with r, duh) |

These actions are quite varied, e.g. writing is applied to software as  
well as reading material. My goal was to summarize tweets, but I keep  
thinking reading all of them is interesting!

**Packages?**

I wondered how many of the tokens correspond to a package name. I  
limited myself to CRAN packages, by using the available.packages()  
function, but one could have a look at the [source code of the  
available package](https://github.com/ropenscilabs/available) to get  
an idea of how to find names of packages from Bioconductor and GitHub.

cran\_pkgs <- as.character(

available.packages(contrib.url('https://cran.r-project.org', 'source'))[,"Package"])

pkg\_tokens <- dplyr::mutate(tokens,

token = gsub("#", "", token)) %>%

dplyr::filter(token %in% cran\_pkgs)

Using the data I’ll look at tweets with the most packages, and most  
frequent packages.

pkg\_tokens %>%

dplyr::group\_by(text) %>%

dplyr::mutate(pkg\_text = paste(toString(token), text)) %>%

dplyr::count(pkg\_text, sort = TRUE) %>%

head(n = 3) %>%

dplyr::pull(pkg\_text)

## [1] "portfolio, blogdown, rmarkdown, knitr, shiny, maps I really like seeing all these #rstats 2019 goals. My own, in order of urgency:\n1) Finish my personal website and online portfolio using blogdown\n2) Get rolling with project workflows, rmarkdown, and knitr \n3) Create shiny apps for custom interactive maps"

## [2] "inference, projects, import, rvest, httr, xml2 My #rstats 2019 goals:\n1. Improve my statistical modeling and inference skills\n2. Develop business literacy and apply it in data analysis projects\n3. Continue to post on my blog (1 post every 2 months)\n4. Learn to import data using DBI, rvest, httr, and xml2"

## [3] "shiny, templates, shiny, shiny, bookdown, shiny My #RStats goals for 2019: \n\n1 Improve shinydashboardPlus, bs4Dash and argonDash .. \n\n2 Release new shiny templates \n3 Open a consulting service for https://t.co/k3PAbxyVMa about shiny \n4 Write an advanced shiny book with bookdown \n#rstats #shiny #consulting https://t.co/Fyc7MhaeW8"

There are false positives, e.g. projects was here meant as a word, not  
a package name. What about the most popular packages among the tweets?

dplyr::count(pkg\_tokens, token, sort = TRUE)

## # A tibble: 66 x 2

## token n

##

## 1 shiny 16

## 2 blogdown 9

## 3 projects 9

## 4 rmarkdown 9

## 5 tidyverse 6

## 6 bookdown 5

## 7 purrr 4

## 8 track 4

## 9 caret 3

## 10 markdown 3

## # ... with 56 more rows

In this table, we get a glimpse at current popular packages, apart from  
“projects”, “track” and “markdown”. If I’m reading the list correctly  
they’re all developed at RStudio!

**Conclusion**

In this post I followed an approach similar to [Jason  
Baik’s](https://github.com/jasonbaik94/rstats-2019-goals/blob/master/analysis.Rmd)  
to summarize tweets about 2019 R goals announced on Twitter: I collected  
tweets with rtweet and then used tidytext and the tidyverse to  
summarize them. Goals often included learning about stuff, building  
packages (find [my list of  
resources](https://masalmon.eu/2017/12/11/goodrpackages/) and don’t miss  
[this offer by Steph de  
Silva](https://twitter.com/StephdeSilva/status/1079871083535167488)),  
and mentions of RStudio packages.

What about my own R goals, that I haven’t tweeted? I have not made any  
list, but have exciting projects at work, and hope to keep  
semi-consistently posting on this blog. In January I’ll also get to  
start 2019 by giving two R talks, [one at R-Ladies  
Paris](https://www.meetup.com/rladies-paris/events/257585346/) and a  
remote one at [ConectaR 2019](https://www.conectar2019.org/#schedule)!  
Happy 2019,