The Algorithm

Mikhail Zhilkin

2017-05-03

```
library(jsonlite)
library(tidyverse)
```

Fetch JSON blob:

```
df <- fromJSON("https://relativio-jccm.rhcloud.com/api/data")
```

```
head(df)
```

```
##                        _id
## 1 590745ea85aa983b6ba47d46
## 2 59077b2d85aa983b6ba47d48
## 3 59077b5b85aa983b6ba47d4a
## 4 59077c6c85aa983b6ba47d4c
## 5 59077c7c85aa983b6ba47d4e
## 6 59077c7d85aa983b6ba47d4f
##                                                                 link
## 1  http://ruletka.se/ads/foretaget-soker-en-personlig-assistent/
## 2                     http://ruletka.se/ads/minnoe-pole-lyubvi/
## 3                    http://ruletka.se/ads/pomogu-vyjti-zamuzh/
## 4  http://ruletka.se/ads/ishhu-muzhchinu-dlya-legkix-otnoshenij/
## 5                    http://ruletka.se/ads/pomogu-vyjti-zamuzh/
## 6 http://ruletka.se/ads/sdam-odnokomnatnuyu-kvartiru-v-spanga-2/
##                                              token                timestamp
## 1 _relativio_fe28e47e-a4e4-4613-8129-aa6dc73bfef5 2017-05-01T14:27:54.095Z
## 2 _relativio_a8e5d856-e0e4-4047-b586-d1a5d0fa1a26 2017-05-01T18:15:09.106Z
## 3 _relativio_421bd4d0-0fb6-4811-8a5c-83937a447b4b 2017-05-01T18:15:55.260Z
## 4 _relativio_23beea6c-9d9a-4dd1-b635-cf893455bda2 2017-05-01T18:20:28.276Z
## 5 _relativio_78ebb224-af96-4fa7-8b9a-f1e6bc2d70af 2017-05-01T18:20:44.289Z
## 6 _relativio_d310098e-29fc-4ba5-9634-613bf5643e60 2017-05-01T18:20:45.726Z
```

Find tokens with multiple unique links:

```
tokens_with_multiple_links <- df %>%
    # valid links & tokens
    filter(
        grepl("^http://ruletka.se/", link),
        grepl("^_relativio_", token)
    ) %>%
    # remove common part
    mutate(
        link = stringr::str_match(link, "^http://ruletka.se/(.*)")[, 2],
        token = stringr::str_match(token, "^_relativio_(.*)")[, 2]
    ) %>%
    # group unique links by token
    group_by(token) %>%
    summarise(links = paste(unique(link), collapse = ",")) %>%
    # only leave tokens with multiple links (column "links"" contains a comma)
    filter(grepl(",", links))
```

```
head(tokens_with_multiple_links)
```

```
## # A tibble: 4 × 2
##                                   token
##                                   <chr>
## 1 0970e09e-2713-4d6a-b337-852991c230d2
## 2 1549e73f-ae8b-4564-b57b-6da179353207
## 3 9e7bd584-daec-40cf-b297-31e939356f81
## 4 ba6108a1-cde1-4759-abd6-3e7952ada2b3
## # ... with 1 more variables: links <chr>
```

This function takes a comma-delimited string with links and turns it into a matrix with all possible link-pairs:

```
str2xref <- function(s) expand.grid(
        A = strsplit(s, ",")[[1]],
        B = strsplit(s, ",")[[1]],
        stringsAsFactors = FALSE
    ) %>% filter(A < B)
```

For example:

```
str2xref("ads/besplatno-parikmaxer/,ads/nuzhen-rabotnik-s-pravami/,ads/sdam-komnatu-25/")
```

```
##                                A                              B
## 1      ads/besplatno-parikmaxer/ ads/nuzhen-rabotnik-s-pravami/
## 2      ads/besplatno-parikmaxer/           ads/sdam-komnatu-25/
## 3 ads/nuzhen-rabotnik-s-pravami/           ads/sdam-komnatu-25/
```

Do it for all strings:

```
do.call(
    rbind,
    lapply(tokens_with_multiple_links$links, str2xref)
) %>% count(A, B) %>% arrange(-n)
```

```
## Source: local data frame [33 x 3]
## Groups: A [11]
##
##                                     A
##                                 <chr>
## 1                   ads/dlya-braka-2/
## 2                   ads/dlya-braka-2/
## 3                   ads/dlya-braka-2/
## 4                   ads/dlya-braka-2/
## 5                   ads/dlya-braka-2/
## 6                   ads/dlya-braka-2/
## 7                   ads/dlya-braka-2/
## 8  ads/individualnyj-podbor-sputnic/
## 9  ads/individualnyj-podbor-sputnic/
## 10 ads/individualnyj-podbor-sputnic/
## # ... with 23 more rows, and 2 more variables: B <chr>, n <int>
```