Playground

Frances Hung

10/21/2017

Playground

```
require(httr)
require(jsonlite)
require(lubridate)
require(dplyr)
require(data.table)
require(tidyr)
require(ggplot2)
```

```
make_dataframe <- function(url) {
    filler <- GET(url)
    filler2 <- rawToChar(filler$content) %>% fromJSON() %>% .[4]
    return(do.call(what = "rbind", args = lapply(filler2, as.data.frame)))
}
```

Reading in Headlines

```
al_jazeera <- make_dataframe("https://newsapi.org/v1/articles?source=al-jazeera-english&sortBy=top&apiK
colnames(al_jazeera)[2] <- "al_jazeera"
bbc <- make_dataframe("https://newsapi.org/v1/articles?source=bbc-news&sortBy=top&apiKey=990d579b0b44403
colnames(bbc)[2] <- "bbc"
breitbart <- make_dataframe("https://newsapi.org/v1/articles?source=breitbart-news&sortBy=top&apiKey=990
colnames(breitbart)[2] <- "breitbart"
reuters <- make_dataframe("https://newsapi.org/v1/articles?source=reuters&sortBy=top&apiKey=990d579b0b44
colnames(reuters)[2] <- "reuters"
nyt <- make_dataframe("https://newsapi.org/v1/articles?source=the-new-york-times&sortBy=top&apiKey=990d5
colnames(nyt)[2] <- "nyt"
```

What proportion of headlines in one source contain a certain key word(s)?

```
topic_contain <- function(list, dataset) {
    p <- TRUE
    for (i in length(list)) {
        new <- grepl(list[i], dataset)
        p <- (p & new)
    }
    sum(p)/10
}
```

Applying the above function to multiple sources

```
topics <- function(dataset, terms) {
    t(as.data.frame(unlist(lapply(terms, topic_contain, dataset))))
}
```

```
topics(reuters$title, list(c("Trump", "JFK"), c("Japan")))
```

```
##                                              [,1] [,2]
## unlist(lapply(terms, topic_contain, dataset))    0    0
```

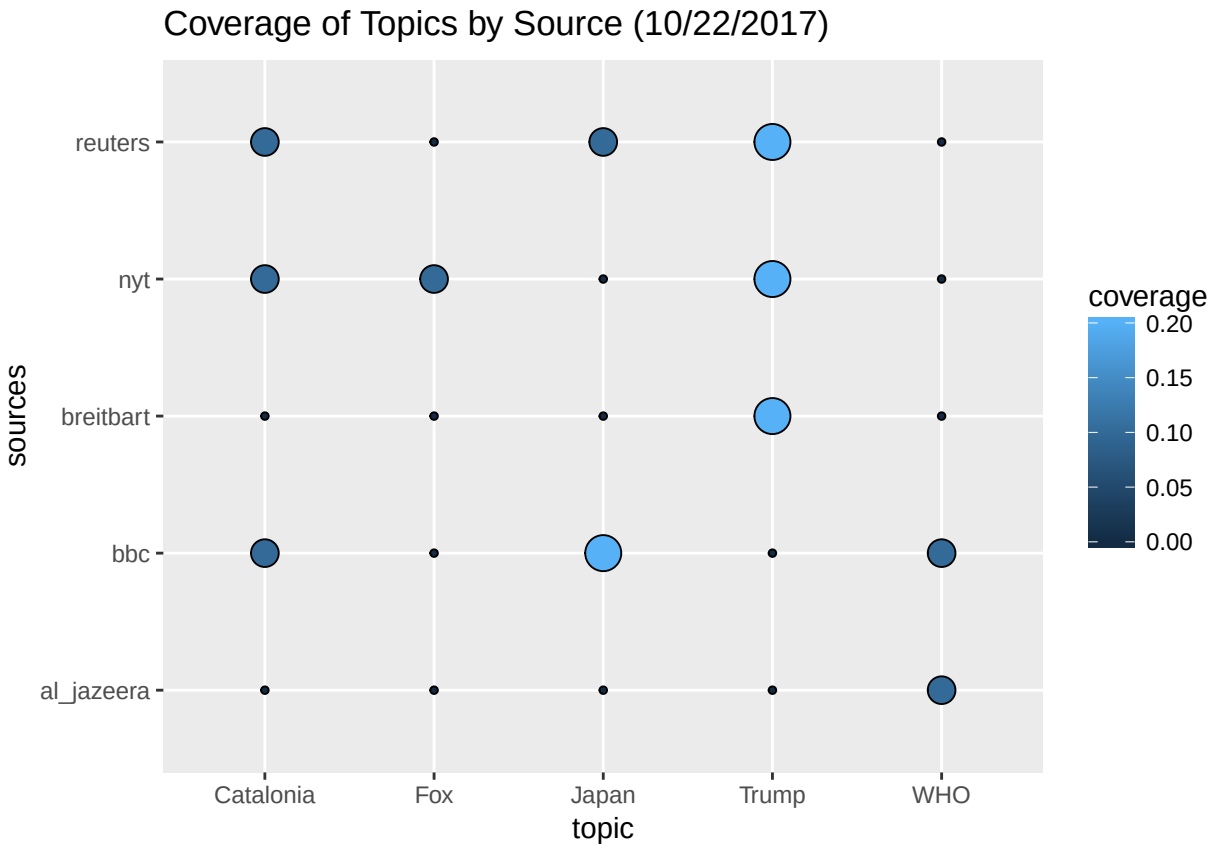
Making our DataFrame

```
key_words <- list(c("Trump"), c("Japan"), "Catalonia", "WHO", "Fox")
news <- c(reuters, bbc, al_jazeera, breitbart, nyt)
test <- as.data.frame(sapply(news, topics, key_words))
test <- test[, !duplicated(colnames(test))]
row.names(test) <- key_words
tops_news <- as.data.frame(t(subset(test, , -c(author, url, urlToImage, description,
    publishedAt))))
tops_news$names <- rownames(tops_news)
tops_news
```

```
##            Trump Japan Catalonia WHO Fox      names
## reuters      0.2   0.1       0.1 0.0 0.0    reuters
## bbc          0.0   0.2       0.1 0.1 0.0        bbc
## al_jazeera   0.0   0.0       0.0 0.1 0.0 al_jazeera
## breitbart    0.2   0.0       0.0 0.0 0.0  breitbart
## nyt          0.2   0.0       0.1 0.0 0.1        nyt
```

Plotting a Bubble Graph

```
tops_news %>% gather(topic, coverage, -names) %>% ggplot(aes(x = topic, y = names)) +
    geom_point(aes(size = coverage, fill = coverage), shape = 21) + guides(size = FALSE) +
    ylab("sources") + ggtitle("Coverage of Topics by Source (10/22/2017)")
```


Coverage of Topics by Source (10/22/2017)
sources
reuters
nyt
breitbart
bbc
al_jazeera
Catalonia
Fox
Japan
Trump
WHO
topic
coverage
0.20
0.15
0.10
0.05
0.00