

Sentiment Analysis

Antonio Osamu Katagiri Tanaka - A01212611@itesm.mx

May 01, 2020

Part 1: paper mining

Load libraries and set custom settings

```
# Clear all objects (from the workspace)
rm(list = ls())

# Strings are not factors
options(stringsAsFactors = F)

# Install and load libraries
library(RISmed)
library(tm)
```

```
## Loading required package: NLP
```

Define the requested query and seek

```
query_colon <-
  "\"electrospinning\"[TIAB] AND (\"NFES\"[TIAB] OR (\"near\"[TIAB] AND \"field\"[TIAB]))"
search_query <- EUutilsSummary(query_colon)

# Let's take a look
summary(search_query)
```

```
## Query:
## "electrospinning"[TIAB] AND ("NFES"[TIAB] OR ("near"[TIAB] AND "field"[TIAB]))
##
## Result count: 66
```

Fetch the data as dataframes

```
records <- EUutilsGet(search_query)
pubmed_data <-
  data.frame(
    'Title' = ArticleTitle(records),
    'Abstract' = AbstractText(records),
    'PID' = ArticleId(records)
  )

# Let's take a look to the 1st search
pubmed_data[1, ]
```

```
##
```

```
## 1 Fiber Lithography: A Facile Lithography Platform Based on Electromagnetic Phase Modulation Using a I
```

```
##
```

```
## 1 Lithography plays a key role in advancing manufacturing as well as the semiconductor industry. Howe
```

```
## PID
```

```
## 1 32297731
```

Process the data

```
# Remove characters : , ; [ ] ( ) from titles and abstracts
pubmed_data$Title <-
  gsub(pattern = "\\.:|,|;|\\[|\\]|\\(|\\)|\\|-",
        replacement = "",
        pubmed_data$Title)
pubmed_data$Abstract <-
  gsub(pattern = "\\.:|,|;|\\[|\\]|\\(|\\)|\\|-",
        replacement = "",
        pubmed_data$Abstract)

# Remove upper case in titles and abstracts
pubmed_data$Title <- tolower(pubmed_data$Title)
pubmed_data$Abstract <- tolower(pubmed_data$Abstract)

# Let's take a look to the 1st search
pubmed_data[1, ]
```

```
##
```

```
## 1 fiber lithography a facile lithography platform based on electromagnetic phase modulation using a h
```

```
##
```

```
## 1 lithography plays a key role in advancing manufacturing as well as the semiconductor industry howev
```

```
##      PID
```

```
## 1 32297731
```

```
# Are there empty abstracts?
```

```
which(pubmed_data$Abstract == "")
```

```
## [1]  6  9 11 14 27 35 49
```

```
# Fetch the words within all abstracts in a dataframe.
```

```
# data frame para guardar las palabras
```

```
word_list <- c()
```

```
#Ciclo para todos los abstracts
```

```
for (i in 1:length(pubmed_data$Abstract)) {
```

```
  #Obtener las palabras como vector en lugar de lista
```

```
  titlePabstract <- paste(pubmed_data$Title[i], pubmed_data$Abstract[i], sep = " ")
```

```
  aux_word <- unlist(strsplit(titlePabstract, " "))
```

```
  #aux_word <- unlist(strsplit(pubmed_data$Abstract[i], " "))
```

```
  #Si el abstract tiene palabras
```

```
  if (length(aux_word) > 0) {
```

```
    #Se juntan las palabras y el PUBMED ID
```

```
    aux_list <- cbind(pubmed_data$PID[i], aux_word)
```

```
    #Se pega este data frame auxiliar al que guarda todo
```

```
    word_list <- rbind(word_list, aux_list)
```

```
  }
```

```
}
```

```
colnames(word_list) <- c("PID", "Word")
```

```
# Let's take a look
```

```
dim(word_list)
```

```
## [1] 11544      2
```

```
# Let's take a look
```

```
head(word_list)
```

```
##      PID      Word
```

```
## [1,] "32297731" "fiber"
```

```
## [2,] "32297731" "lithography"
```

```
## [3,] "32297731" "a"
## [4,] "32297731" "facile"
## [5,] "32297731" "lithography"
## [6,] "32297731" "platform"

# Remove stopwords with tm

# Fetch the English stop_words from tm DB
stop_words <- stopwords(kind = "en")
head(stop_words)

## [1] "i"      "me"      "my"      "myself" "we"      "our"

# Use the indexes to remove stopwords
index_stop_word <- which(word_list[, 2] %in% stop_words)

# Let's take a look
dim(word_list)

## [1] 11544      2

word_list <- word_list[-index_stop_word, ]

# Let's take a look
dim(word_list)

## [1] 7605      2

# Show the 10 most popular words
sort(table(word_list[,2]), decreasing=T)[1:10]

##
## electrospinning      fibers      nanofibers      nearfield      can
##          141          88          68          67          58
##          polymer      3d      electrospun      fiber      using
##          51          49          48          46          44

# Remove duplicated words within each abstract

# Identify each word's abstract origin
word_df <- data.frame(PID=as.numeric(word_list[,1]), Word=word_list[,2],
PIDWord=as.character(apply(word_list, 1, paste, collapse="_")))

# Remove duplicates
dup_index <- duplicated(word_df$PIDWord)
dim(word_df) # Let's take a look

## [1] 7605      3

word_df <- word_df[-which(dup_index),]

# Let's take a look
dim(word_df)

## [1] 5678      3

# Show the 50 most popular words (no duplicates)
sort(table(word_df$Word), decreasing=T)[1:50]

##
## electrospinning      nearfield      can      fibers      applications
##          56          40          32          31          28
##          using      electrospun      field      nanofibers      polymer
##          26          25          25          23          23
##          fiber      nfes      process      method      near
##          20          20          20          18          18
##          technique      potential      substrate      tissue      used
```

##	18	17	17	17	17
##	engineering	fabrication	also	cells	however
##	16	16	15	15	15
##	materials	voltage	fabricated	solution	study
##	15	15	14	14	14
##	based	cell	control	different	high
##	13	13	13	13	13
##	new	paper	use	applied	development
##	13	13	13	12	12
##	direct	low	nm	oxide	patterns
##	12	12	12	12	12
##	properties	results	structures	via	3d
##	12	12	12	12	11

Let's take a look to specific words

```
word_df <- word_df[order(word_df$PID, decreasing=T),]
index_genes <- which(word_df$Word %in% c("pyrolysis", "carbon", "conductivity"))

# Let's take a look
word_df[index_genes, c("PID", "Word")]
```

##		PID	Word
##	171	32236213	carbon
##	198	32236213	pyrolysis
##	674	31763856	conductivity
##	5530	24727667	conductivity
##	6617	22362025	carbon
##	6765	21446719	carbon

Part 2: COVID-19 Public Sentiment Analysis

Text sentiment analyses can be implemented in the identification of misinformation on social media during the coronavirus pandemic. It could be possible to analyse an extensive number of tweets to determine how false information about the coronavirus spreads on social media. Concerns include statements from about hot water eliminating the virus to deceits about past quarantines. While we are trying to contain the virus' spread, misinformation could drain critical resources and set potentially dangerous distractions. Social media has a strong impact in communication during global crisis. Exposing false, misleading and clickbait content can provide a means to explain whether social media can provide insights into social behaviours in real time.