WhyRFoundation - PwC Poland Hackathon

Ugur DAR - Mustafa CAVUS

27 01 2021

```
setwd("C:/Users/gr/Desktop/data_hackathon")
library(readr)
library(stringr)
library(qdap)
library(tm)
library(stringdist)
library(magrittr)
library(hashr)
```

```
tablea <- read.csv("tableA.csv")
tableb <- read.csv("tableB.csv")
train <- read.csv("train.csv")
valid <- read.csv("valid.csv")
```

Exploring

Matched docs

```
matched <- train[which(train$label == 1),]

ilkon_a <- matched[1:10,1]
ilkon_b <- matched[1:10,2]

for(i in 21:30){
  r <-rbind(tablea[which(tablea$id == ilkon_a[i]),],tableb[which(tableb$id == ilkon_b[i]),])
  print(r)
}
```

```
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
```

```
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
## [1] id      title   authors venue   year
## <0 rows> (or 0-length row.names)
```

Venue levels TableA

```
levels(factor(tablea$venue))
```

```
## [1] ""                          "acm trans . database syst ."
## [3] "sigmod conference"         "sigmod record"
## [5] "vldb"                      "vldb j."
```

Venue levels TableB

```
levels(factor(tableb$venue))
```

```
## [1] ""
## [2] "acm sigmod record"
## [3] "acm transactions on database systems ( tods )"
## [4] "international conference on management of data"
## [5] "the vldb journal -- the international journal on very large data bases"
## [6] "very large data bases"
```

Text Manipulation

Text manipulation - TableB

```
doc_id <- tableb[,1]
textb <- NULL
for(i in 1:2294){
  textb[i] <- paste(tableb[i,c(2,3,4,5)],collapse=" ")
}
textb <- str_replace(textb,"NA"," ")
textb <- gsub('\\b\\w{1}\\b','',textb)
textb <- str_replace(textb,"approximate"," ")
textb <- str_replace(textb, "acm transactions on database systems ( tods )","acmsigmodrectrans")
textb <- str_replace(textb,"international conference on management of data","sigmodconference")
textb <- str_replace(textb,"acm sigmod record"  ,"sigmodrecord")
textb <- str_replace(textb,"the vldb journal -- the international journal on very large data bases","vl
textb <- str_replace(textb,"very large data bases","vldb")
textb <- removePunctuation(textb) # nokta ünlem gibi işaretleri siliyor.
textb <- tolower(textb)
textb <- removeWords(textb, stopwords("en"))#ekleri siliyor.
textb <- stripWhitespace(textb) #büyük boşlukları siliyor.
year_b <- gsub(".*(199[0-9]|20[01][0-9]).*","\\1",textb)
textb <- removeNumbers(textb)
df_b <- data.frame(doc_id = doc_id, text = textb,year=year_b)
head(df_b)
```

```
##   doc_id
## 1      0
## 2      1
## 3      2
## 4      3
## 5      4
## 6      5
##
## 1
## 2
## 3                                                                    world wide databaseintegra
## 4                                                                xmlbased information mediation mix
## 5
## 6  cornell jaguar project adding mobility predator phillippe bonnet kyle buza zhiyuan chan victor ch
##   year
## 1 1999
## 2 1999
## 3 1999
## 4 1999
## 5 1999
## 6 1999
```

Text manipulation - TableA

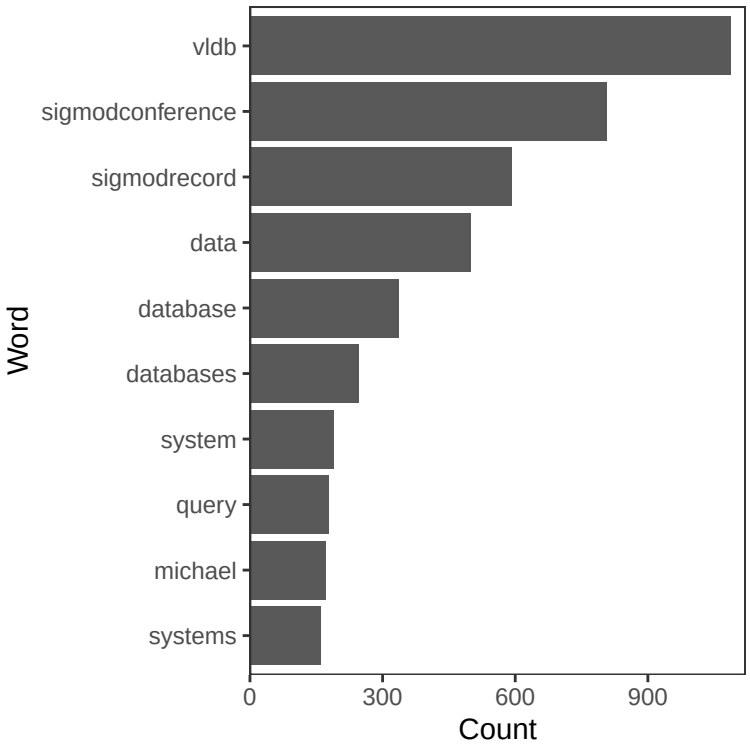
```
doc_id <- tablea[,1]
texta <- NULL
for(i in 1:2616){
  texta[i] <- paste(tablea[i,c(2,3,4,5)],collapse=" ")
```

```
}

texta <- str_replace(texta,"NA"," ")
texta <- gsub('\\b\\w{1}\\b','',texta)
texta <- str_replace(texta,"approximate"," ")
texta <- str_replace(texta, "acm trans . database syst ."," acmsigmodrectrans")
texta <- str_replace(texta,"sigmod conference","sigmodconference")
texta <- str_replace(texta,"sigmod record"  ,"sigmodrecord")
texta <- str_replace(texta,"vldb j.","vldbj")
texta <- removePunctuation(texta) # nokta ünlem gibi işaretleri siliyor.
texta <- tolower(texta)
texta <- removeWords(texta, stopwords("en"))#ekleri siliyor.
texta <- stripWhitespace(texta) #büyük boşlukları siliyor.
year_a <- gsub(".*(199[0-9]|20[01][0-9]).*","\\1",texta)
texta <- removeNumbers(texta)
df_a <- data.frame(doc_id = doc_id, text = texta,year=year_a)
head(df_a)
```

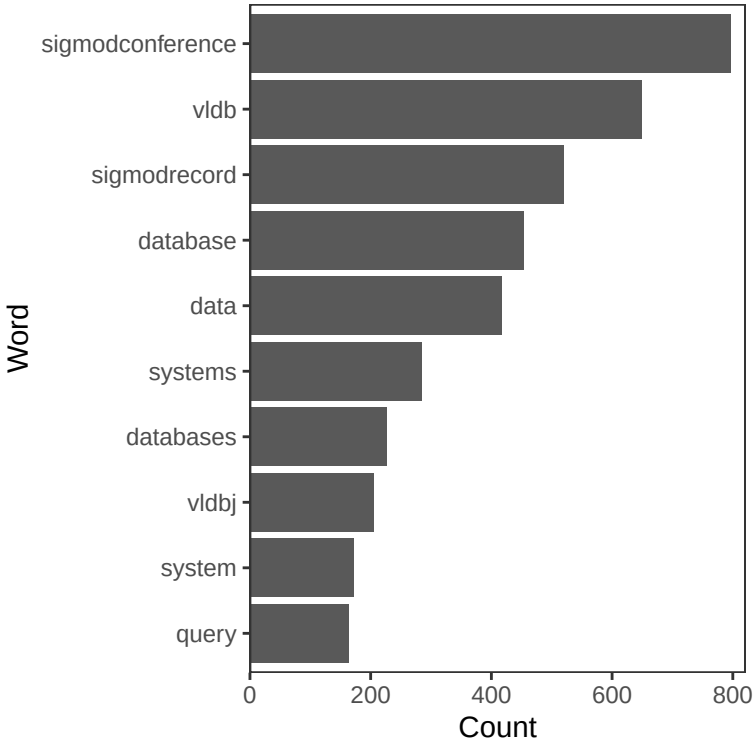
```
##   doc_id
## 1      0
## 2      1
## 3      2
## 4      3
## 5      4
## 6      5
##
## 1                semantic integration environmental models application global information systems de
## 2                     estimation queryresult distribution application paralleljoin load balancing vl
## 3    incremental maintenance nondistributive aggregate functions vldb  themistoklis palpanas richard
## 4 costbased selection path expression processing algorithms objectoriented databases zhaohui tang ge
## 5                                                          benchmarking spatial join operations spat
## 6                                                 efficient geometrybased similarity search d spatial
##   year
## 1 1999
## 2 1996
## 3 2002
## 4 1996
## 5 1995
## 6 1999
```

Visualizing words after text manipulation

```
term_count_a <- freq_terms(texta, 10)
term_count_b <- freq_terms(textb, 10)

plot(term_count_a)
```




```
plot(term_count_b)
```



Text Matching

Train Data

```
n_train <- dim(train)[1]
for(i in 1:n_train){
  sim_mat <-  data.frame(text_sim   = stringsim(df_a[train$ltable_id[i]+1,],
                                    df_b[train$rtable_id[i]+1,],
                                    method = 'jw'))
  sor <- (df_a[train$ltable_id[i]+1,"year"]) == (df_b[train$rtable_id[i]+1,"year"])
  if(sim_mat[2,] <0.79){
    train$den[i] <- 0
  }else
    train$den[i] <-1*sor
}



acc <- NULL
for(i in 1:n_train){
  acc[i] <- train$label[i] == train$den[i]
}

paste("Train set accuracy :",mean(acc))
```

```
## [1] "Train set accuracy : 0.991101523527033"
```

```
# This part more suitable for the data but accuracy is less than stringsim()


# n_train <- dim(train)[1]
# for(i in 1:n_train){
#   sim_mat <-  seq_dist(hash(strsplit(df_a[i,"text"], "\\s+")), hash(strsplit(df_b[i,"text"], "\\s+"))
#   sor <- (df_a[train$ltable_id[i]+1,"year"]) == (df_b[train$rtable_id[i]+1,"year"])
#   if(sim_mat <0.90){
#     train$den[i] <- 0
#   }else
#     train$den[i] <-1*sor
# }
#
# acc <- NULL
# for(i in 1:n_train){
#   acc[i] <- train$label[i] == train$den[i]
# }
#
# paste("Train set accuracy :",mean(acc))
```

Prediction

Valid Data

```
n_valid <- dim(valid)[1]
for(i in 1:n_valid){
  sim_mat <-  data.frame(text_sim   = stringsim(df_a[valid$ltable_id[i]+1,],
```

```
                                    df_b[valid$rtable_id[i]+1,],
                                    method = 'jw'))
  sor <- (df_a[valid$ltable_id[i]+1,"year"]) == (df_b[valid$rtable_id[i]+1,"year"])
  if(sim_mat[2,] <0.79){
    valid$label[i] <- 0
  }else
    valid$label[i] <-1*sor
}
```

```
head(valid)
```

```
##   ltable_id rtable_id label
## 1       141      2211     0
## 2      1074      1849     0
## 3      1367      1815     1
## 4       933      1153     0
## 5      2282       306     0
## 6      1471      1182     0
```

Writing submission file

```
write.csv(valid,"valid-submission.csv",row.names = FALSE)
```