Session 7 Embeddings

Christian Kahmann

12 10 2020

```
# setwd("Your work directory")
options(stringsAsFactors = FALSE)
library(quanteda)
library(text2vec)
```

We use the sotu data once more and create a corpus object

```
textdata <- read.csv("/home/christian/Schreibtisch/Text Mining WS2021/Text Mining Übung/Übung 4/data/so

corpus <- corpus(textdata$text, docnames = textdata$doc_id)

# Build a dictionary of lemmas
lemma_data <- read.csv("/home/christian/Schreibtisch/Text Mining WS2021/Text Mining Übung/Übung 4/resou

# extended stopword list
stopwords_extended <- readLines("/home/christian/Schreibtisch/Text Mining WS2021/Text Mining Übung/Übun

# Create a DTM (may take a while)
corpus_tokens <- corpus %>%
  tokens(remove_punct = TRUE, remove_numbers = TRUE, remove_symbols = TRUE) %>%
  tokens_tolower() %>%
  tokens_replace(lemma_data$inflected_form, lemma_data$lemma) %>%
  tokens_remove(pattern = stopwords_extended, padding = T)

sotu_collocations <- textstat_collocations(corpus_tokens, min_count = 25)
sotu_collocations <- sotu_collocations[1:250, ]

corpus_tokens <- tokens_compound(corpus_tokens, sotu_collocations)
```

```
# Create DTM, but remove terms which occur in less than 0.1% of all documents
DTM <- corpus_tokens %>%
  tokens_remove("") %>%
  dfm() %>%
  dfm_trim(min_docfreq = 0.001, max_docfreq = Inf, docfreq_type = "prop")

# have a look at the number of documents and terms in the matrix
dim(DTM)
```

```
## [1] 21334  3745
```

Create Co-occurrence Matrix

```
library(Matrix)
tcm <- t(DTM)%*%DTM
#sparsity
length(which(tcm==0))/(nrow(tcm)^2)
```

```
## [1] 0.5832599
```

```
v<-colnames(tcm)
```

Now fit the word embeddings using GloVe See: http://nlp.stanford.edu/projects/glove/

```
model = GlobalVectors$new(word_vectors_size=50, vocabulary=v,
                          x_max=10, learning_rate=0.20)
model$fit_transform(tcm,n_iter=25)

wv = t(model$components)
```

Make distance Matrix

```
d = dist2(wv, method="cosine")  #Smaller values means closer
print(dim(d))
```

```
## [1] 3745 3745
```

```
#Pass: w=word, d=dist matrix, n=nomber of close words
findCloseWords = function(w,d,n) {
  words = rownames(d)
  i = which(words==w)
  if (length(i) > 0) {
    res = sort(d[i,])
    print(as.matrix(res[2:(n+1)]))
  }
  else {
    print("Word not in corpus.")
  }
}
```

```
findCloseWords("german",d,10)
```

```
##                          [,1]
## germany             0.1872649
## empire              0.3431974
## berlin              0.3656098
## france              0.4568651
## french              0.4839489
## italy               0.5036886
## emperor             0.5060999
## cotton              0.5193577
## commercial_relation 0.5225802
## prussia             0.5275801
```

```
findCloseWords("germany",d,10)
```

```
##               [,1]
## german   0.1872649
## russia   0.3382942
## berlin   0.3593080
## italy    0.3647804
## belgium  0.3744110
## empire   0.4016698
## european 0.4595688
## france   0.4708657
## sweden   0.4737408
## london   0.4982489
```

```
findCloseWords("environment",d,10)
```

```
##                       [,1]
## clean             0.2304440
## technology        0.3389273
## environmental     0.4015917
## global            0.4026056
## 21st_century      0.4561412
## today's           0.4699573
## ensure            0.4706240
## national_security 0.4739088
## conservation      0.4780374
## gas               0.4813069
```

```
findCloseWords("money",d,10)
```

```
##               [,1]
## currency 0.2517138
## return   0.3132666
## pay      0.3282153
## save     0.3303235
## treasury 0.3361731
## debt     0.3568817
## bond     0.3651405
## expense  0.3693073
## credit   0.3768146
## amount   0.3853728
```

```
findCloseWords("terror",d,10)
```

```
##                 [,1]
## terrorist   0.2473274
## weapon      0.2561861
## iraq        0.2632321
## regime      0.2834938
## middle_east 0.2868423
## threat      0.3320088
## tyranny     0.3374752
## democratic  0.3388474
## democracy   0.3565043
## challenge   0.3740530
```

```
combination<-wv["crisis",,drop=F]+wv["environmental",,drop=F]
distances<-as.vector(dist2(x = combination,y = wv,method = "cosine"))
names(distances)<-colnames(tcm)
sort(distances,decreasing = F)[1:10]
```

```
## environmental        crisis  conservation        energy   environment
##     0.2065497     0.3555052     0.4165459     0.4537158     0.4739067
##      historic      mobilize  dramatically        global      reliable
##     0.4763725     0.4767494     0.4800429     0.4847008     0.4892352
```