RStudio Full DGA Text Analysis for Hana

* Manually cleaned data removing all unnecessary punctuation, whitespace, and numbers.
* Saved cleaned text as a .txt file
* Read the data into RStudio
  + Option1
    - DGA\_raw <- readLines("/Users/connornorton/Desktop/DGA\_full\_clean.txt")
  + Option 2
    - library(readtext)
    - DGA\_raw <- readtext("/Users/connornorton/Desktop/DGA\_full\_clean.txt")
* (OPTIONAL) Convert data into a dataframe
  + First convert the DGA\_string into a matrix, then into a data frame
    - DGA\_matrix <- as.matrix(DGA\_string)
    - DGA\_df <- as.data.frame(DGA\_matrix)
* Add text to a Corpus
  + DGA\_Corpus <- Corpus(VectorSource(DGA\_raw))
* Preprocess the Corpus
  + DGA\_Corpus <- tm\_map(DGA\_Corpus, tolower)
  + DGA\_Corpus <- tm\_map(DGA\_Corpus, removePunctuation)
  + DGA\_Corpus <- tm\_map(DGA\_Corpus, removeNumbers)
  + Removing stopwords
    - library(stopwords)
    - stop\_words <- stopwords("en")
    - stop\_words <- stop\_words[!stop\_words %in% c("can", "cannot", "could", "may", "must", "should", "shall", "would", "are", "be", "able", "to", "need")]
    - DGA\_Corpus <- tm\_map(DGA\_Corpus, removeWords, stop\_words)
* Document Term Matrix
  + Create a Document Term Matrix from the text in the Corpus
    - DGA\_DTM <- DocumentTermMatrix(DGA\_Corpus)
  + Find modal verb frequency using *tm* library
    - library(tm)
    - modal\_verb\_freq <- colSums(as.matrix(DGA\_DTM[, c("can", "cannot", "could", "may", "must", "should", "shall", "would")]))

Text

Description automatically generated with low confidence

* + - Check for modal verb PHRASE frequency as well
      * library(stringr)
      * Convert data into a string
        + DGA\_string <- as.String(DGA\_Corpus)
      * freq\_areAbleTo <- str\_count(DGA\_string, "are able to")
      * freq\_beAbleTo <- str\_count(DGA\_string, "be able to")
      * freq\_needTo <- str\_count(DGA\_string, "need to")
      * freq\_needsTo <- str\_count(DGA\_string, "needs to")
    - Combine all modal verb frequencies into one vector
      * total\_modal\_verb\_freq <- append(modal\_verb\_freq, freq\_areAbleTo)
      * total\_modal\_verb\_freq <- append(total\_modal\_verb\_freq, freq\_beAbleTo)
      * total\_modal\_verb\_freq <- append(total\_modal\_verb\_freq, freq\_needTo)
      * total\_modal\_verb\_freq <- append(total\_modal\_verb\_freq, freq\_needsTo)
      * names(total\_modal\_verb\_freq)[9] <- "are able to"
      * names(total\_modal\_verb\_freq)[10] <- "be able to"
      * names(total\_modal\_verb\_freq)[11] <- "need to"
      * names(total\_modal\_verb\_freq)[12] <- "needs to"
    - Generate Table
      * table\_modal\_verb\_freq <- data.frame(values = unlist(total\_modal\_verb\_freq))

Table

Description automatically generated

* + Find all word frequency
    - library(tm)
    - all\_term\_freq <- colSums(as.matrix(DGA\_DTM))
* (OPTIONAL) Create a Term Document Matrix from the text in the Corpus
  + DGA\_TDM <- TermDocumentMatrix(DGA\_Corpus)
* Create individual word tokens from the Corpus
  + all\_tokens <- text\_tokens(DGA\_Corpus)
* Check the term stats of the Corpus
  + Most Generic
    - term\_stats(DGA\_Corpus)

Table

Description automatically generated

* + After adding filter to remove most common English stop words
    - term\_stats(DGA\_Corpus, subset = !term %in% stopwords\_en)

Table

Description automatically generated

* + - * Can’t remove these stop words because it includes some of Hana’s keywords like “could”, but not “shall” for some reason
  + **After adding filter to remove common stopwords without deleting the modal verbs**
    - **term\_stats(DGA\_Corpus, subset = !term %in% stop\_words)**

Table

Description automatically generated

* Generate word clouds
  + library(wordcloud)
  + Of all terms
    - All Black
      * Only the terms that appear < 21 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 21)
        + min freq = 21 because most of the modal verbs occur 21 or more times in the text

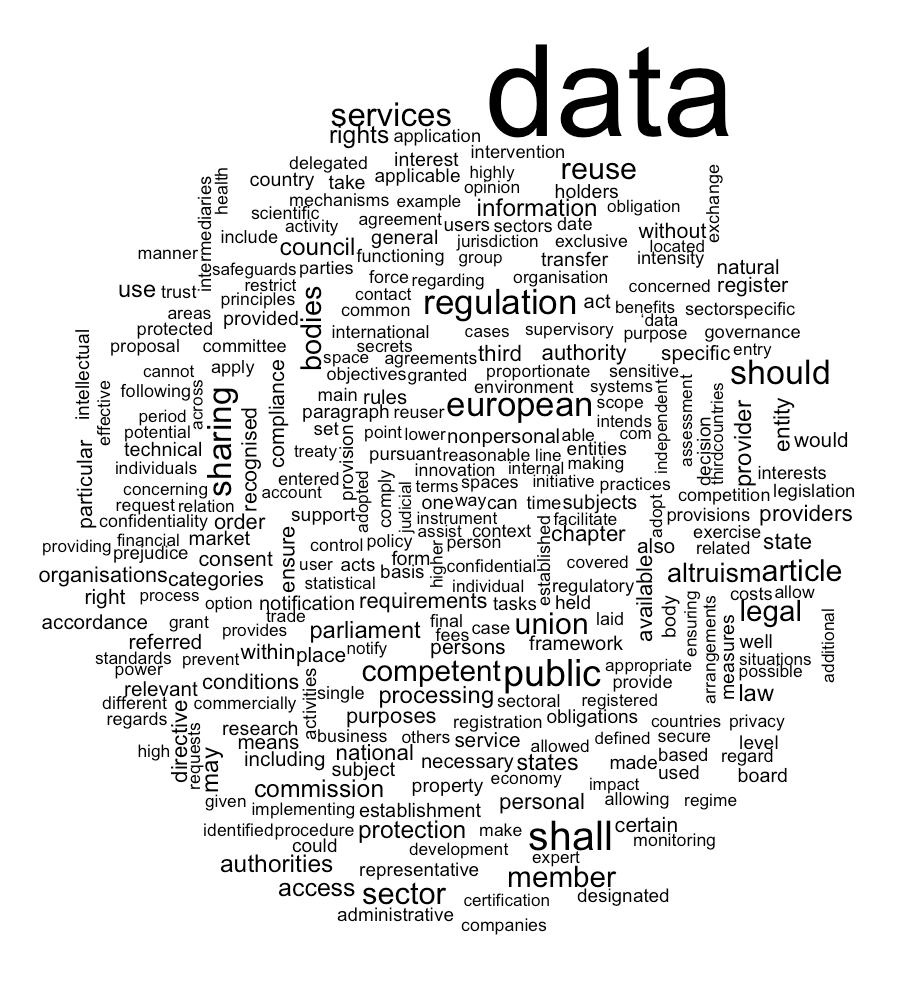
Text

Description automatically generated

* + - * With Custom Stopwords removed, all terms that appear < 21 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 21)



* + - * With Custom Stopwords removed, all terms that appear < 8 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 8)



* + - Color (Spectral)
      * Only the terms that appear < 21 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 21, colors = brewer.pal(11,"Spectral"))

Text

Description automatically generated

* + - * With Custom Stopwords removed, all terms that appear < 21 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 21, colors = brewer.pal(11,"Spectral"))

Text

Description automatically generated

* + - * With Custom Stopwords removed, all terms that appear < 8 times removed
        + wordcloud(names(all\_term\_freq), all\_term\_freq, min.freq = 8, colors = brewer.pal(11,"Spectral"))

Text

Description automatically generated

* + Of only modal verbs
    - Basic (all black)
      * wordcloud(names(total\_modal\_verb\_freq), total\_modal\_verb\_freq, min.freq = 1)

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated Text

Description automatically generated

* + - Color (RdBu)
      * wordcloud(names(total\_modal\_verb\_freq), total\_modal\_verb\_freq, min.freq = 1, colors = brewer.pal(6,"RdBu"))

Logo, company name

Description automatically generated Logo, company name

Description automatically generated Text, logo, company name

Description automatically generated

* + - * wordcloud(names(total\_modal\_verb\_freq), total\_modal\_verb\_freq, min.freq = 1, colors = brewer.pal(8,"RdYlBu"))

Logo, company name

Description automatically generated A picture containing logo

Description automatically generated Logo, company name

Description automatically generated

* + - * wordcloud(names(total\_modal\_verb\_freq), total\_modal\_verb\_freq, min.freq = 1, colors = brewer.pal(9,"Spectral"))

Text, company name

Description automatically generated Logo, company name

Description automatically generated Logo, company name

Description automatically generated

* Libraries used
  + Quanteda
  + Tm
    - To make corpus
  + Corpus
  + Utf8
  + Wordcloud
  + Readtext
  + stopwords
* R License info

This software is distributed under the terms of the GNU General

Public License, either Version 2, June 1991 or Version 3, June 2007.

The terms of version 2 of the license are in a file called COPYING

which you should have received with

this software and which can be displayed by RShowDoc("COPYING").

Version 3 of the license can be displayed by RShowDoc("GPL-3").

Copies of both versions 2 and 3 of the license can be found

at https://www.R-project.org/Licenses/.

A small number of files (the API header files listed in

R\_DOC\_DIR/COPYRIGHTS) are distributed under the

LESSER GNU GENERAL PUBLIC LICENSE, version 2.1 or later.

This can be displayed by RShowDoc("LGPL-2.1"),

or obtained at the URI given.

Version 3 of the license can be displayed by RShowDoc("LGPL-3").