Similarity Calculation

Erin M. Buchanan

9/12/2019

First, words will be selected from the SUBTLEX projects that are: 1) Not stopwords (the, an, of) 2) Three or more characters 3) In the top 10,000 words

We are using the Lg10WF data - this is the log of the word frequency from the subtitle counts. Using log of frequency is advantageous, we can compare the frequencies across datasets, as well as deal with the large skew present in frequency data.

##Library to read excel files  
library(readxl)  
  
##library for stopwords  
library(stopwords)  
  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

##Import the US English Data  
US\_freq <- read\_excel("similarity\_data/SUBTLEXusfrequencyabove1.xls")  
##Import the Dutch Data  
load("similarity\_data/SUBTLEX-NL.cd-above2.Rdata")  
NL\_freq <- subtlex.nl.cdgt2  
rm(subtlex.nl.cdgt2)  
  
##lower case all words  
US\_freq$Word <- tolower(US\_freq$Word)  
NL\_freq$Word <- tolower(NL\_freq$Word)  
  
##Grab the top 10,000 words   
US\_subset <- US\_freq %>% #data frame  
 filter(!Word %in% stopwords(language = "en")) %>% #take out stop words   
 filter(nchar(Word) >= 3) %>% #words greater than or equal 3  
 arrange(Lg10WF) %>% #sort by Log10WF   
 top\_n(10000) #Take the top 10k

## Selecting by Lg10CD

NL\_subset <- NL\_freq %>% #data frame  
 filter(!Word %in% stopwords(language = "nl")) %>% #take out stop words   
 filter(nchar(Word) >= 3) %>% #words greater than or equal 3  
 arrange(Lg10WF) %>% #sort by Log10WF   
 top\_n(10000) #Take the top 10k

## Selecting by Lg10CD

##these libraries will be combined with translate R  
##this service is not free, uses google's API  
##will work with the university to see if we have  
##a service already for this type of task   
#library(translateR)