Vulgar language on the web across World Englishes - Part 1: data preparation

Anonymous

2025-04-05

Contents

1 Intro

This document shows an analysis that was performed with the aim of finding differences in swearing across geographically distinct varieties of English around the world based on the GloWbe corpus.

install packages

```
# install packages
install.packages("tidyverse")
install.packages("quanteda")
install.packages("here")
install.packages("udpipe")
install.packages("future")
install.packages("furrr")
install.packages("stringi")
install.packages("parallel")
install.packages("usethis")
install.packages("data.table")
install.packages("tokenizers")
```

load packages and set options

```
# load packages
library(data.table)
library(tidyverse)
library(quanteda)
library(here)
library(stringi)
```

```
library(parallel)
library(usethis)
library(data.table)
library(quanteda)
library(future)
library(furrr)
library(tokenizers)
```

2 Load data

define paths

```
# list files
fls <- list.files("D:/corpora/GloWbE",
                  pattern = "txt",
                  full.names = T,
                  recursive = T,
                  include.dirs = T)
# inspect
head(fls)
```

function for preparing data

```
preptxt <- function(fls, var) {
  fln  <- fls[stringr::str_detect(fls, var)]
  # WARNING! FOR TESTING PURPOSES
  #fln <- fln[1:2]
  txt <- sapply(fln, function(x){
    x <- scan(x, what = "char", quote = "", skipNul = T) %>%
      paste0(collapse = " ")  %>%
      stringr::str_replace_all("([#]{2,3}\\d+)", "~~~\\1") %>%
      stringr::str_split("~~~") %>%
      unlist()
    }) %>%
    unlist()
  crp  <- names(txt)  %>% stringr::str_remove_all(".*GloWbE/") %>% stringr::str_remove_all("/Texts.*")
  sfl  <- names(txt)  %>% stringr::str_remove_all(".*/")
  fl  <- txt  %>% stringr::str_remove_all(" .*") %>% stringr::str_remove_all(".txt.*")
  df <- data.frame(crp, fl, sfl, txt) %>%
    dplyr::filter(txt != "")
  # WRANING FOR TESTING PURPOSES!
  base::saveRDS(df, file = here::here("test", paste0(var, "df.rda", collapse = "", sep = "")))
  #base::saveRDS(df, file = here::here("data", paste0(var, "dftest.rda", collapse = "", sep = "")))
}
```

```
#fls   <- fls[1:3]
#preptxt(fls, var = "Australia") # Australia
audf  <- base::readRDS(file = here::here("test", "Australiadftest.rda"))
# inspect
head(audf); nrow(audf)
```

3 Processing

```
# processing
preptxt(fls, var = "Australia") # Australia
preptxt(fls, var = "Bangladesh") # Bangladesh
preptxt(fls, var = "Canada") # Canada
preptxt(fls, var = "GB-Blog") # GB-Blog
preptxt(fls, var = "GB-General") # GB-General
preptxt(fls, var = "Ghana") # Ghana
preptxt(fls, var = "Hong Kong") # Hong Kong
preptxt(fls, var = "India") # India
preptxt(fls, var = "Ireland") # Ireland
preptxt(fls, var = "Jamaica") # Jamaica
preptxt(fls, var = "Kenya") # Kenya
preptxt(fls, var = "Malaysia") # Malaysia
preptxt(fls, var = "New Zealand") # New Zealand
preptxt(fls, var = "Nigeria") # Nigeria
preptxt(fls, var = "Pakistan") # Pakistan
preptxt(fls, var = "Philippines") # Philippines
preptxt(fls, var = "Singapore") # Singapore
preptxt(fls, var = "South Africa") # South Africa
preptxt(fls, var = "Sri Lanka") # Sri Lanka
preptxt(fls, var = "Tanzania") # Tanzania
preptxt(fls, var = "US-Blog") # US-Blog
preptxt(fls, var = "US-General") # US-General
```

check

```
audf  <- base::readRDS(file = here::here("test", "Australiadf.rda"))
# inspect
head(audf); nrow(audf)
```

```
##                                                                                                 crp
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt2 Australia
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt3 Australia
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt4 Australia
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt5 Australia
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt6 Australia
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt7 Australia
##                                                                                                   fl
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt2  ##703
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt3 ##1000
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt4 ##1004
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt5 ##1100
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt6 ##1400
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt7 ##1500
##
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt2 w_au_g01.t
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt3 w_au_g01.t
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt4 w_au_g01.t
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt5 w_au_g01.t
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt6 w_au_g01.t
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt7 w_au_g01.t
##
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt2
```

```
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt3
being as a source of stress . Most of us typically perceive stress as a ... <p> In terms of losing weigh
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt4 ##1004 <h
up of a girl is XX . An XO baby is outwardly a girl , but her cells only have one copy of the X chromos
sex chromosomes and 1 sex chromosome ( commonly called the X or the Y chromosome ) <p> Each ' set ' of
question : What are chromosomes ? <p> Chromosomes can also be thought of as similar to a ball of wool -
chromosome is also responsible for differences in female identicals . When a female foetus is created ;
20 cells ) . If by chance one identical twin ' silences ' the X chromosome that came from Dad 's sperm
identical . This is due to something called ' chimerism ' - when an individual is composed of 2 genetica
identical twins are chimeras ; the rise in IVF is considered to be a contributing factor . A report has
copy , but some should be available as electronic versions -- it depends on what type of service the re
87 <p> Department of Psychology USA . This e-mail address is being protected from spambots . You need Ja
sex twin sample . The data were examined with reference to psychobiological and evolutionary perspectiv
2 months before the co-twin 's death , 1-2 months following the co-twin 's death and currently . A Grie
2 months following the loss , and currently . Information on physical symptoms was available from the S
twin ( retrospective twin group ) ; a second @ @ @ @ @ @ @ @ @ @ . Consistent with psychobiological the
mail address is being protected from spambots . You need JavaScript enabled to view it <p> Contemporary
the mourner 's memory and emotions . A complementary perspective is offered by archetypal psychology ,
mail address is being protected from spambots . You need JavaScript enabled to view it <p> We @ @ @ @ @
26 . <p> Department UK . This e-mail address is being protected from spambots . You need JavaScript enab
depth about their understanding and experience of twinship . Participants were selected who had a rich
year study was based on individual interviews of over 200 bereaved MZ and DZ adult twins . Its purpose
analysis . <h> Question : Do you have any resources for twins who do n't get along with each other ? <p
authored another book Uniting Psychology and Biology : Integrative Perspectives on Human Development wh
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt5
third of the land mass . Yet when it comes time to doll out the money each year for the federal road bu
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt6
for-a-good-reason Italian food can be found on the Croatian coast . <p> But because it has none of the
season where prices are low , the weather is good , and you will find very few tourists roaming around
point , but no matter where you go in Buenos Aires , will you most likely turn to your travel companion
sized hole in your wallet . <h> Caribbean = Southern Thailand <p> A beautiful Thai beach on the island
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/Australia/Texts/w_au_g01.txt7
facing wall ( or North facing wall if in the Southern Hemisphere ) , which is painted black and made of
e glass - this is special glass which has a lower rate of heat transmission , similar in effect to doub
14 inches for brick , 12-18 for concrete , 8-12 " for adobe or other earth material and at least 6 inch
e glass for a Trombe wall . Do you have data ? Even with the latest , high-transmittance versions of th
potable @ @ @ @ @ @ @ @ @ @ . am considering all passive solar technologies , may mix with a micro phot
scaled : less water , less heat , less energy consumption . also looking for less complexity in activit
```

```
## [1] 81680
```

Clean text

write function to clean text

```
cleantxt <- function(dfs){
  sapply(dfs, function(x) {
  nmfl <- stringr::str_remove_all(x, ".*/") %>% stringr::str_remove_all("df.rda")
  base::readRDS(file = x)  %>%
    # differentiate between EFF (institution) and eff (fuck)
  dplyr::mutate(txt = stringr::str_replace_all(txt, " EFF", " EFFentity")) %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
```

```
                ctxt = stringr::str_squish(ctxt, remove_separators = F))  %>%
  base::saveRDS(df, file = here::here("data", paste0(nmfl, "dfc.rda", collapse = "", sep = "")))
}) }
```

apply function

```
dfs <- list.files(here::here("data"),
                  pattern = "df.rda",
                  full.names = T,
                  recursive = T,
                  include.dirs = T)
# inspect
head(dfs)
```

```
dfs <- dfs[1]
cleantxt(dfs)
```

```
df <- base::readRDS(file = here::here("test", "Australiadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Australiadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Bangladeshdf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Bangladeshdfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Canadadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Canadadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "GB-Blogdf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "GB-Blogdfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "GB-Generaldf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "GB-Generaldfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Ghanadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Ghanadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Hong Kongdf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Hong Kongdfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Indiadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Indiadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Irelanddf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Irelanddfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Jamaicadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
```

```
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Jamaicadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Kenyadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Kenyadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Malaysiadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Malaysiadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "New Zealanddf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "New Zealanddfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Nigeriadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Nigeriadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Pakistandf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Pakistandfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Philippinesdf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Philippinesdfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Singaporedf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Singaporedfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "South Africadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "South Africadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Sri Lankadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Sri Lankadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "Tanzaniadf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "Tanzaniadfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "US-Blogdf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
```

```
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "US-Blogdfc.rda"))
```

```
df <- base::readRDS(file = here::here("test", "US-Generaldf.rda"))  %>%
  dplyr::mutate(ctxt = tolower(txt)) %>%
  dplyr::mutate(ctxt = iconv(ctxt, to = "ASCII"),
                ctxt = stringr::str_remove(ctxt, ".*? "),
                ctxt = stringr::str_replace_all(ctxt, "<.*?>", " "),
                ctxt = stringr::str_replace_all(ctxt, "[:punct:]", " "),
                ctxt = stringr::str_squish(ctxt))
# save
base::saveRDS(df, file = here::here("test", "US-Generaldfc.rda"))
```

4 Regex

define regex lists

The regex list represents is based on:

List of Bad Words, February 2025. http://www.noswearing.com/dictionary/.

BannedWordList.com - a resource for web administrators, March 2013. http://www.bannedwordlist.com/.

McEnery, Anthony. 2006. Swearing in English: Bad Language, Purity and Power from 1586 to the Present. New York: Routledge.

Thelwall, Mike. 2008. "Fk Yea I Swear: Cursing and Gender in MySpace." Corpora 3 (1): 83–107. doi:10.3366/E1749503208000087.

Coats, S. (2021). 'Bad language' in the Nordics: Profanity and gender in a social media corpus. Acta Linguistica Hafniensia, 53(1), 22–57. https://doi.org/10.1080/03740463.2021.1871218

Love, R. (2021). Swearing in informal spoken English: 1990s–2010s. Text & Talk, 41(5-6), 739-762.

After reviewing the items deemed vulgar in the above publications, we decided which to include as we did not consider all elements in the publications as vulgar. The items we thus deemed as vulgar after reviewing are listed below.

arse, arsehole, ass, asshole, bastard, beaner, bellend, bimbo, bitch, bloody, bollock, boner, bonk, boob, bugger, bullshit, butt, butthead, butthole, chink, cock, coon, crap, cum, cunt, damn, darkie, dick, dike, dildo, dipshit, dork, eff, fag, fanny, fart, feck, frig, fuck, gash, gook, hell, hussy, idiot, jackass, jap, jerk, jiss, jug, kike, knocker, lesbo, minger, moron, motherfucker, muff, nonce, nympho, pecker, pedo, pikey, pimp, piss, poofter, prick, puke, pussy, queef, retard, shag, shit, shite, skank, slag, slut, sod, spastic, tit, tosser, tranny, turd, twat, wank, whore, online (such as wtf, lmao, etc)

The regular expression list below is designed to capture these and variants of these elements.

Vector of regular expressions for detecting vulgar language and obfuscations

```
patterns <- c(
  # "arsehole/s"
  "\\b(dumb|stupid|lazy|worthless|useless|brain|dead|jack)*[a@4äáâ]r[s5$§z][e3€ëéê][h]?[o0øöóõ]*[l1£][e3

  # "ass/asshole/s"
  "\\b(dumb|stupid|lazy|worthless|useless|brain|dead|jack)*[a@4äáâ][s5$§z]{2,}[h]?[o0øöóõ]*[l1|£][e3€ëé

  # "bitch"
```

```
"\\b[8ß|3][i1!|ïíì]+[a@4äáå]*[t7+†][c¢©(][h](es|ez|ing|ed)*\\b",

# "bastard"
"\\b[8ß|3][a@4äáå][s5$§z][t7+†][a@4äáå]r[d][o]*(s|z)*\\b",

# "beaner"
"\\b[8ß|3][e3€ëéê][a@4äáå]n[e3€ëéê]r(s|z)*\\b",

# "bellend"
"\\b[8ß|3][e3€ëéê]ll[e3€ëéê]nd(s|z)*\\b",

# "bimbo"
"\\b[8ß|3][1!|ïíì]mb[oO0øöóõ](s|z)*\\b",

# "bloody"
"\\bbl[oO0øöóõ]{2,}d[iy¥](ed)*\\b",

# "bollocks"
"\\b[8ß|3][oO0øöóõ]ll[oO0øöóõiïíì][xc¢©(k|<{(]+[s5$§z]?\\b",

# "boner"
"\\bboner[s]*\\b",

# "bonk"
"\\b[8ß|3][oO0øöóõ]n[k|<{(](in|ing)*\\b",

# "boobs"
"\\b[8ß|3][oO0øöóõ]{2,}[b8ß|3][ie]*[s5$§z]?\\b",

# "bugger"
"\\b[8ß|3][u|µüúû]gg[e3€ëéê]r(ing|s|z)*\\b",

# "bullshit"
"\\b[b8ß|3][u|µüúû]ll[s5$§z]h[1!|ïíì]+[t7+†]*\\b",

# "butt"
"\\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b",

# "damn"
"\\b(god)*damn\\b",

# "darkie"
"\\bdarki(es)*\\b",

# "dike"
"\\b(bull)*d[iy]*ke(s|z)*\\b",

# "dildo"
"\\bdildo(s|z)*\\b",

# "dork"
"\\bdork(s|z)*\\b",
```

```
# "eff"
"\\beff(ing|in|ed|d)*\\b",


# "fanny"
"\\bfann(y|ies)+\\b",

# "fart"
"\\bfart(s|z|ing|in|ed)*\\b",

# "frig"
"\\bfrig(g|gin|ging|ged|gs)*\\b",

# Detects "fuck" variations
"\\b(cluster|head|mother|motha|mutha|mada|cock|mom|mum|daddy|father|sister|brother)*[f=f][uµüúû|@a4äá
"\\b[f=f][c¢©(*k<{(uµüúû*]+(ing|er|a|ed|ers|az)*\\b",

# Detects "fuck" variations as 'f'
"\\bf\\b",

# "gash"
"\\bgash\\b",

# "gook"
"\\bg[o]*ok(s|z)*\\b",

# "idiot"
"\\bidiot(s|z)*\\b",

# "jackass"
"\\bjacka[s5$§z][s5$§z]\\b",

# "jap"
"\\bjap[zs]*\\b",

# "jerk"
"\\bjerk(s|z|in|ing|ed)*\\b",

# "jiss"
"\\bji[s5$§z][s5$§z]+\\b",

# "jug"
"\\bjug[g]*[s5$§z]*\\b",

# "shit"
"\\b[(dip)]*[s5$§z]h[1!|ïíì]+[t7+†]*(ing|e|in|er|a|ed|ers|az|s|z)*\\b",

# Online variants

# Knock the f*** out
"\\bktfo\\b",

# Shut the f*** up
```

```
"\\bstfu\\b",

# Get the f*** out
"\\bgtfo\\b",

# Not giving a f***
"\\bngaf\\b",

# Don't give a f***
"\\bdgaf\\b",

# For f***'s sake
"\\bffs\\b",

# F*** my life
"\\bfml\\b",

# Oh my f***ing god
"\\bomfg\\b",

# As f***
"\\baf\\b",

# The f***
"\\btf\\b",

# What the f***
"\\bwtf\\b",

# Laughing my ass off
"\\blmao\\b",

# Laughing my f***ing ass off
"\\blmfao\\b",

# Rolling on floor laughing
"\\brofl\\b",

# "chink"
"\\bch[i]*nk[zs]*\\b",

# "coon"
"\\bcoon[zs]*\\b",

# "crap"
"\\b[bull]*crap(ping|ped|s|z|pin)*\\b",

# "cum"
"\\bcum(ming)*\\b",

# "cock"
"\\bc[oO0øöóõ][c¢©]+[(k|<{(|x]+(suck|sak|suk)*[k]*(er|ers|a|az|as)*\\b",
```

```
# "cunt"
"\\b[kc¢©(][u|µüúû]*nt[zs]*\\b",

# "dick"
"\\b[d][1!|ïíì][c¢©(][xk|<{(][(head)]*[zs]*\\b",

# derogatory term for homosexual
"\\b[f|=f][a@4äáâe]g[g]*[ioa]*[t]*[zs]*\\b",

# "hoe"
"\\bh[o0øöóõ][e3€ëéê][zs]*\\b",

# "hore"
"\\bh[o0øöóõ]r[e3€ëéê]*[zs]*\\b",

# "kike"
"\\b[k|<{(][i1!ïíìy][k|<{(][e3€ëéê][zs]*\\b",

# racial slur
"\\bn[i1!ïíì]gg[e3€ëéê|@a4äáâ][r]*[zs]*\\b",

# "knob"
"\\bknob[(head)]*[zs]*\\b",

# "lesbo"
"\\blesbo[sz]*\\b",

# "minger"
"\\bming[(a|er)]+(s|z)*\\b",

# "moron"
"\\bm[o|u]ron(ic|s|z)*\\b",

# "muff"
"\\bmuff\\b",

# "nonce"
"\\bnonce\\b",

# "nympho"
"\\bnympho\\w*\\b",

# "pecker"
"\\bp[e3€ëéê]ck[ae]?[rs]*\\b",

# "pedo"
"\\bp[e3€ëéê]do[philf]*[e]*[zs]*\\b",

# "pikey
"\\bpik(i|is|ey|ies|eys|eyz|iez|iz)+\\b",

# "pimp"
"\\bpimp(s|ing|in|z|ed)*\\b",
```

```
# "piss"
"\\bp[i1!|ïíì][s5$§z][s5$§]+(in|ing|er|a|ers|erz)*\\b",

# "poofter"
"\\bpooft(er|ers|as|az)+\\b",

# "prick"
"\\bprick[zs]*\\b",

# "puke"
"\\bpuk[e]*(s|z|ing|ed)*\\b",

# "pussy"
"\\bp[u|µüúû][s5$§z][s5$§z][@a4äáå]*[y¥][zs]*\\b",

# "queef"
"\\bqu[e]+[a]*f(s|z|ing|ed)*\\b",

# "shag"
"\\bshag(ging|gin|ged)*\\b",

# "skank"
"\\bskank[yzs]*\\b",

# "slag"
"\\bslag[zs]*\\b",

# "slut"
"\\b[s5$§z]l[uµüúû][t7+†](i|y)*[zs]*\\b",

# "sod"
"\\bsod[d]*[sz]*(ing)*\\b",

# "spastic"
"\\bspast(ic|ics|icz)*\\b",

# "retard"
"\\bretard[zs]*\\b",

# "tits"
"\\btit[t]*(i|ies|ay|ays|ayz)*\\b",

# "tosser"
"\\btosser[sz]*\\b",

# "tranny"
"\\btr[a@4äáå]nn(y|ies|iez)+\\b",

# "turd"
"\\bturd[sz]*\\b",

# "twat"
"\\b[t7+†]w[a@4äáå][t7+†][zs]*\\b",
```

```
  # "wank"
  "\\bw[a@4äáå]n[k|<{(](z|er|ers|ing|az|a|ed)*\\b",

  # "whore"
  "\\b(cam|man|m)*wh[o0øöóõ]?r[e3€ëéê]*(d|s|z|ing)*\\b"

)
```

5 KWIC

```
# Australia
t0 <- Sys.time()
df <- base::readRDS(file = here::here("test", "Australiadfc.rda"))

kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Australia")

# save
base::saveRDS(kwic_results, file = here::here("test", "Australia_kwic_results.rda"))
t1 <- Sys.time()
t1-t0
```

```
## Time difference of 56.87991 secs
```

```
# inspect
head(kwic_results, 20); names(table(kwic_results$keyword)); nrow(kwic_results)
```

```
##    docname                              pre keyword
## 1   text14              but to take a large     hoe
## 2   text19            get within a monkey s    fart
## 3   text20               when i first saw a     jug
## 4   text20             10 59 am cleaning ur    butt
## 5   text20               do the same to you    butt
## 6   text20           shelly do you wash your    butt
## 7   text20             hands do you dry your    butt
## 8   text20           the water on your dirty    butt
## 9   text20                    9 2009 2 01 pm    butt
## 10  text20      food industry to wash their   butts
## 11  text20          and then water on their   butts
## 12  text20         have only a tissue wiped    butt
## 13  text20           03 pm re water washing   butts
## 14  text20                    9 2009 3 53 pm    butt
## 15  text20    i wash up properly afterwards    butt
## 16  text20           hand only and wash your    butt
## 17  text20       method 1 when washing your    butt
## 18  text20      over your knees holding the     jug
## 19  text20 liberal amounts of water between    butt
## 20  text20   convert and definitely wash my    butt
##                                  post
## 1               and a shovel also and
## 2         of being a musician working
```

```
## 3                of water in an indian
## 4              with water is much more
## 5    smearing is not cleaning washing
## 6               with your hands do you
## 7               after you ve washed it
## 8            the other two options are
## 9             washing again i see that
## 10         with their hands however if
## 11           and then wash their hands
## 12              if i down there posted
## 13       in 3rd world countries where
## 14             washing hmmm i still do
## 15 washing is definitely cleaner than
## 16               with your left its an
## 17           hold the water whether it
## 18            tap with your right hand
## 19               crack you can feel it
## 20     after a dump normally speaking
##                                                                                  pattern
## 1                                                                \\bh[o0øöóõ][e3€ëéê][zs]*\\b
## 2                                                                  \\bfart(s|z|ing|in|ed)*\\b
## 3                                                                       \\bjug[g]*[s5$§z]*\\b
## 4  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 5  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 6  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 7  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 8  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 9  \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 10 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 11 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 12 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 13 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 14 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 15 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 16 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 17 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 18                                                                      \\bjug[g]*[s5$§z]*\\b
## 19 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
## 20 \\b[b8ß3][uµüúû][t7+†][t7+†][sz]*(face|head|wit|whipe|hole|h)*[hl]*[sz]*\\b
##       corpus
## 1  Australia
## 2  Australia
## 3  Australia
## 4  Australia
## 5  Australia
## 6  Australia
## 7  Australia
## 8  Australia
## 9  Australia
## 10 Australia
## 11 Australia
## 12 Australia
## 13 Australia
## 14 Australia
```

```
## 15 Australia
## 16 Australia
## 17 Australia
## 18 Australia
## 19 Australia
## 20 Australia
```

```
##   [1] "$300b"         "$800b"         "$af"           "$f"
##   [5] "=f"            "3=c"           "3000b"         "3003"
##   [9] "30035"         "3008"          "300b"          "4551"
##  [13] "8003"          "8008"          "a$$hole"       "a$$holes"
##  [17] "a$$sholes"     "af"            "alt+f"         "ar$ehole"
##  [21] "ar$ehole$"     "arsehole"      "arseholes"     "asshole"
##  [25] "assholes"      "assshole"      "azzl"          "bl00dy"
##  [29] "bloodied"      "bloody"        "bloooody"      "boner"
##  [33] "boners"        "bullcrap"      "bullsh1t"      "butt"
##  [37] "butthead"      "butthole"      "butts"         "c=c"
##  [41] "chink"         "chinks"        "clusterfuck"   "cnt"
##  [45] "cock"          "cocker"        "cockers"       "control+f"
##  [49] "coon"          "coons"         "crap"          "crapped"
##  [53] "crapping"      "craps"         "ctrl+f"        "cum"
##  [57] "cumming"       "cunt"          "cunts"         "d1ckheads"
##  [61] "damn"          "darkies"       "dgaf"          "dike"
##  [65] "dikes"         "dildo"         "dildos"        "dork"
##  [69] "dorks"         "dyke"          "dykes"         "eff"
##  [73] "effed"         "effin"         "effing"        "f"
##  [77] "f$"            "f$135"         "f$250"         "f$3000"
##  [81] "f$ck"          "f$ckin"        "f+"            "f+ck"
##  [85] "f+cking"       "f+g"           "f="            "f=1"
##  [89] "f=14"          "f=3"           "f=c"           "f=focal"
##  [93] "f=ma"          "f3ck"          "faaaking"      "facking"
##  [97] "fag"           "fagg"          "faggot"        "faggots"
## [101] "faggott"       "fagot"         "fagots"        "fags"
## [105] "fake"          "faked"         "faker"         "fakers"
## [109] "faking"        "fannies"       "fanny"         "fart"
## [113] "farted"        "farting"       "farts"         "fc"
## [117] "fca"           "fcaa"          "fcc"           "fcca"
## [121] "fccc"          "fcing"         "fck"           "fcked"
## [125] "fcking"        "fcu"           "fcuk"          "fcuked"
## [129] "fcuking"       "fe=c"          "feck"          "feckers"
## [133] "fecking"       "feek"          "feg"           "fegg"
## [137] "fego"          "fek"           "feke"          "ffs"
## [141] "fk"            "fkc"           "fkers"         "fking"
## [145] "fkr"           "fml"           "foak"          "fockers"
## [149] "foecke"        "fok"           "fook"          "fooked"
## [153] "fooker"        "fookers"       "frig"          "frigg"
## [157] "frigged"       "friggin"       "frigging"      "fu"
## [161] "fu$"           "fu$king"       "fua"           "fuc"
## [165] "fucing"        "fuck"          "fucked"        "fuckee"
## [169] "fucker"        "fuckers"       "fuckhead"      "fucking"
## [173] "fuckwit"       "fued"          "fuek"          "fuk"
## [177] "fuke"          "fuked"         "fukers"        "fuking"
## [181] "fukk"          "fukking"       "fuku"          "fuu"
## [185] "fuuuck"        "fuuuuuck"      "fuuuuuuck"     "gash"
```

```
## [189] "gdp=c+i+g+x"   "goddamn"       "gook"          "gooks"
## [193] "gtfo"          "hoe"           "hoes"          "hoess"
## [197] "home=c"        "hor"           "hore"          "hors"
## [201] "idiot"         "idiots"        "jackass"       "jap"
## [205] "japs"          "jerk"          "jerked"        "jerkin"
## [209] "jerking"       "jerkings"      "jerkins"       "jerks"
## [213] "jizz"          "jug"           "jugg"          "juggs"
## [217] "jugs"          "kike"          "knob"          "knobhead"
## [221] "knobs"         "knt"           "knts"          "kunt"
## [225] "kuntz"         "kyke"          "lesbo"         "lesbos"
## [229] "lmao"          "lmfao"         "m2=c"          "manwhore"
## [233] "manwhores"     "manwhoring"    "meta+f"        "minga"
## [237] "mingas"        "minge"         "minger"        "moron"
## [241] "moronic"       "morons"        "motherfcker"   "motherfucker"
## [245] "motherfuckers" "motherfucking" "muff"          "mwhr"
## [249] "mwhrs"         "nigga"         "niggas"        "niggaz"
## [253] "nigge"         "nigger"        "niggers"       "nonce"
## [257] "norway=f"      "nympho"        "nymphoes"      "nympholeptic"
## [261] "nymphomania"   "nymphomaniac"  "nymphomaniacs" "omfg"
## [265] "or=f"          "p1ss"          "peck"          "pecker"
## [269] "peckers"       "pecks"         "pedo"          "pedofile"
## [273] "pedofiles"     "pedophile"     "pedophiles"    "pedos"
## [277] "pi$$ed"        "pi$$es"        "pimp"          "pimped"
## [281] "pimpin"        "pimping"       "pimps"         "pis$ed"
## [285] "piss"          "pisser"        "pissin"        "pissing"
## [289] "pooftas"       "poofter"       "poofters"      "prick"
## [293] "pricks"        "puk"           "puke"          "puked"
## [297] "pukes"         "puking"        "pukings"       "pussy"
## [301] "queef"         "r2=c"          "retard"        "retards"
## [305] "rofl"          "sh1"           "sh1t"          "sh1te"
## [309] "sh1ts"         "shag"          "shag=whoops"   "shagged"
## [313] "shagging"      "skank"         "skanks"        "skanky"
## [317] "slag"          "slags"         "slut"          "sluts"
## [321] "sluty"         "sod"           "sodding"       "soding"
## [325] "sods"          "spastic"       "spastics"      "start+f"
## [329] "stfu"          "t=f"           "tf"            "tit"
## [333] "titties"       "tosser"        "tossers"       "trannies"
## [337] "tranny"        "turd"          "turds"         "twat"
## [341] "twats"         "twok=400k"     "v=c"           "view=uk"
## [345] "wank"          "wanka"         "wanked"        "wanker"
## [349] "wankers"       "wanking"       "wh0re"         "wh0res"
## [353] "whore"         "whored"        "whores"        "whoring"
## [357] "whr"           "whre"          "wtf"           "y=c"
## [361] "zealand=f"
```

```
## [1] 39119
```

```
# Bangladesh
df <- base::readRDS(file = here::here("test", "Bangladeshdfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Bangladesh")
# save
base::saveRDS(kwic_results, file = here::here("test", "Bangladesh_kwic_results.rda"))
```

```
# Canada
df <- base::readRDS(file = here::here("test", "Canadadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Canada")
# save
base::saveRDS(kwic_results, file = here::here("test", "Canada_kwic_results.rda"))

# GB-Blog
df <- base::readRDS(file = here::here("test", "GB-Blogdfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "GB-Blog")
# save
base::saveRDS(kwic_results, file = here::here("test", "GBBlog_kwic_results.rda"))

# GB-General
df <- base::readRDS(file = here::here("test", "GB-Generaldfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "GB-General")
# save
base::saveRDS(kwic_results, file = here::here("test", "GBGeneral_kwic_results.rda"))

# Ghana
df <- base::readRDS(file = here::here("test", "Ghanadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Ghana")
# save
base::saveRDS(kwic_results, file = here::here("test", "Ghana_kwic_results.rda"))

# Hong Kong
df <- base::readRDS(file = here::here("test", "Hong Kongdfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Hong Kong")
# save
base::saveRDS(kwic_results, file = here::here("test", "HongKong_kwic_results.rda"))

# India
df <- base::readRDS(file = here::here("test", "Indiadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "India")
# save
base::saveRDS(kwic_results, file = here::here("test", "India_kwic_results.rda"))
```

```r
# Ireland
df <- base::readRDS(file = here::here("test", "Irelanddfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Ireland")
# save
base::saveRDS(kwic_results, file = here::here("test", "Ireland_kwic_results.rda"))

# Jamaica
df <- base::readRDS(file = here::here("test", "Jamaicadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Jamaica")
# save
base::saveRDS(kwic_results, file = here::here("test", "Jamaica_kwic_results.rda"))

# Kenya
df <- base::readRDS(file = here::here("test", "Kenyadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Kenya")
# save
base::saveRDS(kwic_results, file = here::here("test", "Kenya_kwic_results.rda"))

# Malaysia
df <- base::readRDS(file = here::here("test", "Malaysiadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Malaysia")
# save
base::saveRDS(kwic_results, file = here::here("test", "Malaysia_kwic_results.rda"))

# New Zealand
df <- base::readRDS(file = here::here("test", "New Zealanddfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "New Zealand")
# save
base::saveRDS(kwic_results, file = here::here("test", "NewZealand_kwic_results.rda"))

# Nigeria
df <- base::readRDS(file = here::here("test", "Nigeriadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Nigeria")
# save
base::saveRDS(kwic_results, file = here::here("test", "Nigeria_kwic_results.rda"))
```

```r
# Pakistan
df <- base::readRDS(file = here::here("test", "Pakistandfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Pakistan")
# save
base::saveRDS(kwic_results, file = here::here("test", "Pakistan_kwic_results.rda"))

# Philippines
df <- base::readRDS(file = here::here("test", "Philippinesdfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Philippines")
# save
base::saveRDS(kwic_results, file = here::here("test", "Philippines_kwic_results.rda"))

# Singapore
df <- base::readRDS(file = here::here("test", "Singaporedfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Singapore")
# save
base::saveRDS(kwic_results, file = here::here("test", "Singapore_kwic_results.rda"))

# South Africa
df <- base::readRDS(file = here::here("test", "South Africadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "South Africa")
# save
base::saveRDS(kwic_results, file = here::here("test", "SouthAfrica_kwic_results.rda"))

# Sri Lanka
df <- base::readRDS(file = here::here("test", "Sri Lankadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Sri Lanka")
# save
base::saveRDS(kwic_results, file = here::here("test", "SriLanka_kwic_results.rda"))

# Tanzania
df <- base::readRDS(file = here::here("test", "Tanzaniadfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "Tanzania")
# save
base::saveRDS(kwic_results, file = here::here("test", "Tanzania_kwic_results.rda"))
```

```
# US-Blog
df <- base::readRDS(file = here::here("test", "US-Blogdfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "US-Blog")
# save
base::saveRDS(kwic_results, file = here::here("test", "USBlog_kwic_results.rda"))

# US-General
df <- base::readRDS(file = here::here("test", "US-Generaldfc.rda"))
kwic_results <- quanteda::kwic(quanteda::tokens(stringi::stri_split_fixed(df$ctxt, " ")), pattern = pat
  as.data.frame() %>%
  dplyr::select(-from, -to) %>%
  dplyr::mutate(corpus = "US-General")
# save
base::saveRDS(kwic_results, file = here::here("test", "USGeneral_kwic_results.rda"))
```

```
head(df)
```

```
##                                                                                                 c
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt2 US-
General
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt3 US-
General
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt4 US-
General
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt5 US-
General
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt6 US-
General
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt7 US-
General
##                                                                                                   fl
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt2  ##101
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt3  ##600
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt4  ##800
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt5 ##1001
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt6 ##1200
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt7 ##1201
##
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt2 w_us_g01
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt3 w_us_g01
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt4 w_us_g01
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt5 w_us_g01
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt6 w_us_g01
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt7 w_us_g01
##
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt2 ##101 <h
boiled cubed potatoes and cook everything until crisp , adding in some sliced onions and pepper about ha
flavored Spam ) . I like to add some spicy mayonnaise and Swiss cheese to mine , sandwiching it with an
start on melting . <p> Add some griddled onions to the mix , replace the Wonderbread with rye , and you
Melt . <p> Head down to NOLA , and you may find some of our Spam-endowed brethren place hot spam on a s
free breading if you 'd like ) , sandwiched around slices of crisp bacon , pepperjack cheese , and a su
```

```
duper secret special sauce ( hint : the recipe 's here ) . <h> Spam : It 's What 's For Dinner <p> If y
for-dinner does n't have to be refined to island cuisine . Do n't believe me ? Take a few of these guys
cooked spaghetti in there and mix it all around to form a rich , peppery , eggy , Spamalicious sauce . I
chi Fried Rice . Fried Spam cubes , chopped kimchi , kimchi juice , and day old rice fried up with some
abilities are endless .
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt3
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt4
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt5
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt6
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt7
potassium foods during the day if you plan to go out for dinner . Request a small glass and no refills i
potassium vegetables such as green pepper and onion . To cut down on sodium , potassium and phosphorus
filled muffins are all good options , as well as low-potassium fruits such as pineapple , grapes and app
ounce serving per meal or the amount indicated on your individual meal plan . ( Three ounces is the size
friendly side dishes include steamed rice , buttered noodles or pasta , a small green salad with low-
sodium dressing and coleslaw . Low-potassium vegetables are also a good side order . These include green
##
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt2 spam hack
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt3
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt4
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt5
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt6
## C:/Users/Martin/OneDrive/Dokumente/Corpora/CorporaHDD/GloWbE/US-General/Texts/w_us_g01.txt7
```

6 Outro

```
sessionInfo()
```

```
## R version 4.4.2 (2024-10-31 ucrt)
## Platform: x86_64-w64-mingw32/x64
## Running under: Windows 11 x64 (build 26100)
##
## Matrix products: default
##
##
## locale:
## [1] LC_COLLATE=English_United States.utf8
## [2] LC_CTYPE=English_United States.utf8
## [3] LC_MONETARY=English_United States.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=English_United States.utf8
##
## time zone: Australia/Brisbane
## tzcode source: internal
##
## attached base packages:
## [1] parallel  stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] tokenizers_0.3.0  furrr_0.3.1       future_1.34.0     usethis_3.1.0
##  [5] stringi_1.8.7     here_1.0.1        quanteda_4.2.0    lubridate_1.9.4
##  [9] forcats_1.0.0     stringr_1.5.1     dplyr_1.1.4       purrr_1.0.4
```

```
## [13] readr_2.1.5       tidyr_1.3.1       tibble_3.2.1      ggplot2_3.5.1
## [17] tidyverse_2.0.0   data.table_1.17.0
##
## loaded via a namespace (and not attached):
##  [1] generics_0.1.3    lattice_0.22-6    listenv_0.9.1     hms_1.1.3
##  [5] digest_0.6.37     magrittr_2.0.3    evaluate_1.0.3    grid_4.4.2
##  [9] timechange_0.3.0  bookdown_0.42     fastmap_1.2.0     rprojroot_2.0.4
## [13] Matrix_1.7-1      stopwords_2.3     scales_1.3.0      codetools_0.2-20
## [17] cli_3.6.4         rlang_1.1.5       parallelly_1.43.0 munsell_0.5.1
## [21] withr_3.0.2       yaml_2.3.10       tools_4.4.2       tzdb_0.5.0
## [25] colorspace_2.1-1  fastmatch_1.1-6   globals_0.16.3    vctrs_0.6.5
## [29] R6_2.6.1          lifecycle_1.0.4   fs_1.6.5          pkgconfig_2.0.3
## [33] pillar_1.10.1     gtable_0.3.6      glue_1.8.0        Rcpp_1.0.14
## [37] xfun_0.52         tidyselect_1.2.1  rstudioapi_0.17.1 knitr_1.50
## [41] SnowballC_0.7.1   htmltools_0.5.8.1 rmarkdown_2.29    compiler_4.4.2
```