USAID st. louis

Deon, Akua, Gerardo, Sam

4/22/2020

## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

#install needed packages (only run this once)  
#install.packages('tm')  
#install.packages('tidytext')  
#install.packages('tidyr')  
#install.packages('dplyr')  
#install.packages('NLP')  
#install.packages('ggplot2')  
#install.packages('wordcloud')  
#install.packages('topicmodels')  
#install.packages('quanteda')  
#install.packages('tm')  
#install.packages('SnowballC')  
#install.packages('corpus')  
#install.packages('textdata')

#Library the packages  
library(tm)

## Loading required package: NLP

library(tidytext)

## Warning: package 'tidytext' was built under R version 3.6.2

library(tidyr)  
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

library(NLP)  
library(ggplot2)

##   
## Attaching package: 'ggplot2'

## The following object is masked from 'package:NLP':  
##   
## annotate

library(wordcloud)

## Loading required package: RColorBrewer

library(topicmodels)

## Warning: package 'topicmodels' was built under R version 3.6.2

library(quanteda)

## Package version: 2.0.1

## Parallel computing: 2 of 4 threads used.

## See https://quanteda.io for tutorials and examples.

##   
## Attaching package: 'quanteda'

## The following objects are masked from 'package:tm':  
##   
## as.DocumentTermMatrix, stopwords

## The following objects are masked from 'package:NLP':  
##   
## meta, meta<-

## The following object is masked from 'package:utils':  
##   
## View

library(SnowballC)

## Warning: package 'SnowballC' was built under R version 3.6.2

library(corpus)

## Warning: package 'corpus' was built under R version 3.6.2

#Running the Jaccard Index  
setwd("~/Desktop/USAID Final Project")  
data <- Corpus(DirSource("LoisTxfiles"))  
  
dfmat <- dfm(corpus\_subset(corpus(data)),  
remove\_punct = TRUE, remove = stopwords("english"))  
(tstat1 <- textstat\_simil(dfmat, method = "jaccard", margin = "documents"))

## textstat\_simil object; method = "jaccard"  
## 1.txt 10.txt 11.txt 12.txt 13.txt 14.txt 15.txt 16.txt 17.txt  
## 1.txt 1.0000 0.0543 0.0478 0.0681 0.0387 0.0700 0.0563 0.0920 0.0814  
## 10.txt 0.0543 1.0000 0.0586 0.0707 0.0615 0.1303 0.0744 0.0334 0.0505  
## 11.txt 0.0478 0.0586 1.0000 0.0517 0.0510 0.0926 0.0502 0.0507 0.0431  
## 12.txt 0.0681 0.0707 0.0517 1.0000 0.2182 0.0888 0.0598 0.0418 0.0734  
## 13.txt 0.0387 0.0615 0.0510 0.2182 1.0000 0.0695 0.0397 0.0402 0.0480  
## 14.txt 0.0700 0.1303 0.0926 0.0888 0.0695 1.0000 0.1043 0.0602 0.0650  
## 15.txt 0.0563 0.0744 0.0502 0.0598 0.0397 0.1043 1.0000 0.0393 0.0406  
## 16.txt 0.0920 0.0334 0.0507 0.0418 0.0402 0.0602 0.0393 1.0000 0.0879  
## 17.txt 0.0814 0.0505 0.0431 0.0734 0.0480 0.0650 0.0406 0.0879 1.0000  
## 18.txt 0.0930 0.0553 0.0662 0.0772 0.0515 0.1003 0.0623 0.0854 0.1157  
## 19.txt 0.0959 0.0809 0.0820 0.0882 0.0587 0.1545 0.0696 0.0773 0.0984  
## 2.txt 0.0764 0.0832 0.0577 0.0889 0.0622 0.0855 0.0569 0.0851 0.1382  
## 20.txt 0.0913 0.0547 0.0535 0.0676 0.0493 0.0841 0.0578 0.1043 0.1123  
## 21.txt 0.0813 0.0752 0.0609 0.0620 0.0510 0.0679 0.0751 0.0870 0.1020  
## 22.txt 0.1119 0.0595 0.0420 0.0719 0.0520 0.0669 0.0665 0.0758 0.0674  
## 23.txt 0.0747 0.0828 0.0574 0.0885 0.0619 0.0851 0.0549 0.0834 0.1388  
## 24.txt 0.0823 0.0511 0.0448 0.0667 0.0490 0.0506 0.0562 0.0766 0.0891  
## 25.txt 0.0826 0.0707 0.0603 0.0781 0.0441 0.1161 0.0594 0.0641 0.0774  
## 26.txt 0.0873 0.0420 0.0440 0.0649 0.0407 0.0670 0.0366 0.0939 0.1350  
## 27.txt 0.0692 0.0439 0.0512 0.0452 0.0287 0.0521 0.0521 0.0663 0.0419  
## 28.txt 0.0681 0.0777 0.0671 0.0566 0.0502 0.1728 0.0814 0.0439 0.0551  
## 29.txt 0.0837 0.0639 0.0272 0.0647 0.0470 0.0450 0.0443 0.0791 0.1085  
## 3.txt 0.0663 0.0620 0.0491 0.1049 0.0732 0.0756 0.0920 0.0775 0.0820  
## 30.txt 0.0916 0.0749 0.0412 0.0971 0.0787 0.0744 0.0566 0.0822 0.1196  
## 4.txt 0.0662 0.0447 0.0480 0.0638 0.0557 0.0487 0.0370 0.0827 0.0756  
## 5.txt 0.0823 0.0505 0.0646 0.0762 0.0526 0.0714 0.0569 0.0688 0.0899  
## 6.txt 0.0629 0.0498 0.0482 0.0761 0.0561 0.0492 0.0556 0.0755 0.0594  
## 7.txt 0.0597 0.0554 0.0590 0.0740 0.0520 0.0679 0.0419 0.0739 0.0579  
## 8.txt 0.0610 0.0693 0.0409 0.0408 0.0418 0.0652 0.0591 0.0455 0.0497  
## 9.txt 0.0771 0.0616 0.0344 0.0519 0.0321 0.0487 0.0408 0.0633 0.0602  
## 18.txt 19.txt 2.txt 20.txt 21.txt 22.txt 23.txt 24.txt 25.txt  
## 1.txt 0.0930 0.0959 0.0764 0.0913 0.0813 0.1119 0.0747 0.0823 0.0826  
## 10.txt 0.0553 0.0809 0.0832 0.0547 0.0752 0.0595 0.0828 0.0511 0.0707  
## 11.txt 0.0662 0.0820 0.0577 0.0535 0.0609 0.0420 0.0574 0.0448 0.0603  
## 12.txt 0.0772 0.0882 0.0889 0.0676 0.0620 0.0719 0.0885 0.0667 0.0781  
## 13.txt 0.0515 0.0587 0.0622 0.0493 0.0510 0.0520 0.0619 0.0490 0.0441  
## 14.txt 0.1003 0.1545 0.0855 0.0841 0.0679 0.0669 0.0851 0.0506 0.1161  
## 15.txt 0.0623 0.0696 0.0569 0.0578 0.0751 0.0665 0.0549 0.0562 0.0594  
## 16.txt 0.0854 0.0773 0.0851 0.1043 0.0870 0.0758 0.0834 0.0766 0.0641  
## 17.txt 0.1157 0.0984 0.1382 0.1123 0.1020 0.0674 0.1388 0.0891 0.0774  
## 18.txt 1.0000 0.1100 0.1049 0.1095 0.0903 0.0870 0.1057 0.0699 0.0981  
## 19.txt 0.1100 1.0000 0.1056 0.1051 0.0861 0.0823 0.1039 0.0961 0.1091  
## 2.txt 0.1049 0.1056 1.0000 0.1032 0.1224 0.0730 0.9906 0.0882 0.0912  
## 20.txt 0.1095 0.1051 0.1032 1.0000 0.0888 0.0827 0.1041 0.0763 0.0993  
## 21.txt 0.0903 0.0861 0.1224 0.0888 1.0000 0.0818 0.1219 0.0758 0.0931  
## 22.txt 0.0870 0.0823 0.0730 0.0827 0.0818 1.0000 0.0727 0.0767 0.0754  
## 23.txt 0.1057 0.1039 0.9906 0.1041 0.1219 0.0727 1.0000 0.0867 0.0893  
## 24.txt 0.0699 0.0961 0.0882 0.0763 0.0758 0.0767 0.0867 1.0000 0.0873  
## 25.txt 0.0981 0.1091 0.0912 0.0993 0.0931 0.0754 0.0893 0.0873 1.0000  
## 26.txt 0.1012 0.0962 0.1274 0.1136 0.0987 0.0907 0.1291 0.0975 0.1025  
## 27.txt 0.0547 0.0709 0.0687 0.0684 0.0694 0.0727 0.0684 0.0494 0.0764  
## 28.txt 0.0720 0.1508 0.0617 0.0809 0.0599 0.0697 0.0614 0.0429 0.0878  
## 29.txt 0.0889 0.0685 0.1311 0.0809 0.0878 0.0633 0.1306 0.0727 0.0672  
## 3.txt 0.0798 0.0920 0.0904 0.1036 0.0744 0.0717 0.0885 0.0686 0.0794  
## 30.txt 0.0963 0.0860 0.1573 0.1103 0.0950 0.0671 0.1553 0.0807 0.0855  
## 4.txt 0.0643 0.0665 0.0971 0.0978 0.0633 0.0625 0.0954 0.0571 0.0683  
## 5.txt 0.0780 0.0837 0.1043 0.0881 0.1028 0.0775 0.1039 0.0652 0.0680  
## 6.txt 0.0698 0.0618 0.0786 0.0754 0.0918 0.0648 0.0783 0.0543 0.0919  
## 7.txt 0.0905 0.0703 0.0774 0.0742 0.0626 0.0846 0.0770 0.0637 0.0734  
## 8.txt 0.0618 0.0601 0.0581 0.0518 0.0671 0.0431 0.0561 0.0536 0.0788  
## 9.txt 0.0828 0.0578 0.0730 0.0786 0.0767 0.0670 0.0727 0.0664 0.0833  
## 26.txt 27.txt 28.txt 29.txt 3.txt 30.txt 4.txt 5.txt 6.txt  
## 1.txt 0.0873 0.0692 0.0681 0.0837 0.0663 0.0916 0.0662 0.0823 0.0629  
## 10.txt 0.0420 0.0439 0.0777 0.0639 0.0620 0.0749 0.0447 0.0505 0.0498  
## 11.txt 0.0440 0.0512 0.0671 0.0272 0.0491 0.0412 0.0480 0.0646 0.0482  
## 12.txt 0.0649 0.0452 0.0566 0.0647 0.1049 0.0971 0.0638 0.0762 0.0761  
## 13.txt 0.0407 0.0287 0.0502 0.0470 0.0732 0.0787 0.0557 0.0526 0.0561  
## 14.txt 0.0670 0.0521 0.1728 0.0450 0.0756 0.0744 0.0487 0.0714 0.0492  
## 15.txt 0.0366 0.0521 0.0814 0.0443 0.0920 0.0566 0.0370 0.0569 0.0556  
## 16.txt 0.0939 0.0663 0.0439 0.0791 0.0775 0.0822 0.0827 0.0688 0.0755  
## 17.txt 0.1350 0.0419 0.0551 0.1085 0.0820 0.1196 0.0756 0.0899 0.0594  
## 18.txt 0.1012 0.0547 0.0720 0.0889 0.0798 0.0963 0.0643 0.0780 0.0698  
## 19.txt 0.0962 0.0709 0.1508 0.0685 0.0920 0.0860 0.0665 0.0837 0.0618  
## 2.txt 0.1274 0.0687 0.0617 0.1311 0.0904 0.1573 0.0971 0.1043 0.0786  
## 20.txt 0.1136 0.0684 0.0809 0.0809 0.1036 0.1103 0.0978 0.0881 0.0754  
## 21.txt 0.0987 0.0694 0.0599 0.0878 0.0744 0.0950 0.0633 0.1028 0.0918  
## 22.txt 0.0907 0.0727 0.0697 0.0633 0.0717 0.0671 0.0625 0.0775 0.0648  
## 23.txt 0.1291 0.0684 0.0614 0.1306 0.0885 0.1553 0.0954 0.1039 0.0783  
## 24.txt 0.0975 0.0494 0.0429 0.0727 0.0686 0.0807 0.0571 0.0652 0.0543  
## 25.txt 0.1025 0.0764 0.0878 0.0672 0.0794 0.0855 0.0683 0.0680 0.0919  
## 26.txt 1.0000 0.0881 0.0531 0.0716 0.0823 0.1046 0.0850 0.0915 0.0814  
## 27.txt 0.0881 1.0000 0.0612 0.0476 0.0507 0.0597 0.0494 0.0747 0.0853  
## 28.txt 0.0531 0.0612 1.0000 0.0478 0.0510 0.0532 0.0476 0.0571 0.0703  
## 29.txt 0.0716 0.0476 0.0478 1.0000 0.0722 0.1650 0.0534 0.0620 0.0605  
## 3.txt 0.0823 0.0507 0.0510 0.0722 1.0000 0.0912 0.0719 0.0873 0.0678  
## 30.txt 0.1046 0.0597 0.0532 0.1650 0.0912 1.0000 0.0791 0.0958 0.0778  
## 4.txt 0.0850 0.0494 0.0476 0.0534 0.0719 0.0791 1.0000 0.0759 0.0778  
## 5.txt 0.0915 0.0747 0.0571 0.0620 0.0873 0.0958 0.0759 1.0000 0.0777  
## 6.txt 0.0814 0.0853 0.0703 0.0605 0.0678 0.0778 0.0778 0.0777 1.0000  
## 7.txt 0.0757 0.0671 0.0447 0.0670 0.0640 0.0782 0.0527 0.0822 0.0787  
## 8.txt 0.0510 0.0280 0.0408 0.0476 0.0731 0.0479 0.0427 0.0587 0.0616  
## 9.txt 0.0778 0.0412 0.0497 0.0696 0.0697 0.0610 0.0606 0.0795 0.0690  
## 7.txt 8.txt 9.txt  
## 1.txt 0.0597 0.0610 0.0771  
## 10.txt 0.0554 0.0693 0.0616  
## 11.txt 0.0590 0.0409 0.0344  
## 12.txt 0.0740 0.0408 0.0519  
## 13.txt 0.0520 0.0418 0.0321  
## 14.txt 0.0679 0.0652 0.0487  
## 15.txt 0.0419 0.0591 0.0408  
## 16.txt 0.0739 0.0455 0.0633  
## 17.txt 0.0579 0.0497 0.0602  
## 18.txt 0.0905 0.0618 0.0828  
## 19.txt 0.0703 0.0601 0.0578  
## 2.txt 0.0774 0.0581 0.0730  
## 20.txt 0.0742 0.0518 0.0786  
## 21.txt 0.0626 0.0671 0.0767  
## 22.txt 0.0846 0.0431 0.0670  
## 23.txt 0.0770 0.0561 0.0727  
## 24.txt 0.0637 0.0536 0.0664  
## 25.txt 0.0734 0.0788 0.0833  
## 26.txt 0.0757 0.0510 0.0778  
## 27.txt 0.0671 0.0280 0.0412  
## 28.txt 0.0447 0.0408 0.0497  
## 29.txt 0.0670 0.0476 0.0696  
## 3.txt 0.0640 0.0731 0.0697  
## 30.txt 0.0782 0.0479 0.0610  
## 4.txt 0.0527 0.0427 0.0606  
## 5.txt 0.0822 0.0587 0.0795  
## 6.txt 0.0787 0.0616 0.0690  
## 7.txt 1.0000 0.0601 0.0741  
## 8.txt 0.0601 1.0000 0.0740  
## 9.txt 0.0741 0.0740 1.0000

#Run the Text Analysis  
library(tidytext, pos=15)  
library(SnowballC)  
library(tm, pos=16)  
library(ggplot2)  
  
library(NLP, pos=16)  
library(tm, pos=16)  
library(dplyr)  
  
library(textdata)  
library(topicmodels)  
library(tidyr)  
library(tm)  
library(SnowballC)  
b <- Corpus(DirSource("LoisTxfiles"))  
b <- tm\_map(b, stripWhitespace)  
b <- tm\_map(b, removeNumbers)  
b <- tm\_map(b, removePunctuation)  
b <- tm\_map(b,content\_transformer(tolower))  
b<- tm\_map(b, removeWords, c(stopwords("english"),"]-"))  
  
dtmc<- DocumentTermMatrix(b)  
dtmc

## <<DocumentTermMatrix (documents: 30, terms: 4628)>>  
## Non-/sparse entries: 9688/129152  
## Sparsity : 93%  
## Maximal term length: 27  
## Weighting : term frequency (tf)

termsc <- Terms(dtmc)  
head(terms)

##   
## 1 new("standardGeneric", .Data = function (x, ...)   
## 2 standardGeneric("terms"), generic = structure("terms", package = "stats"),   
## 3 package = "stats", group = list(), valueClass = character(0),   
## 4 signature = "x", default = new("derivedDefaultMethod", .Data = function (x,   
## 5 ...)   
## 6 UseMethod("terms"), target = new("signature", .Data = "ANY",

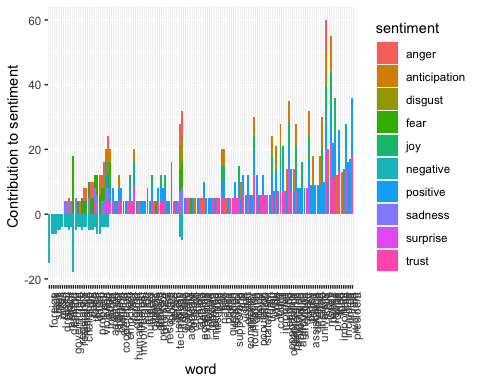
library(tidytext)  
library(dplyr)  
dtm\_tdc <- tidy(dtmc)  
dtm\_tdc

## # A tibble: 9,688 x 3  
## document term count  
## <chr> <chr> <dbl>  
## 1 1.txt acknowledged 1  
## 2 1.txt affected 1  
## 3 1.txt africa 2  
## 4 1.txt aid 1  
## 5 1.txt already 1  
## 6 1.txt also 2  
## 7 1.txt always 2  
## 8 1.txt american 1  
## 9 1.txt amid 1  
## 10 1.txt anna 1  
## # … with 9,678 more rows

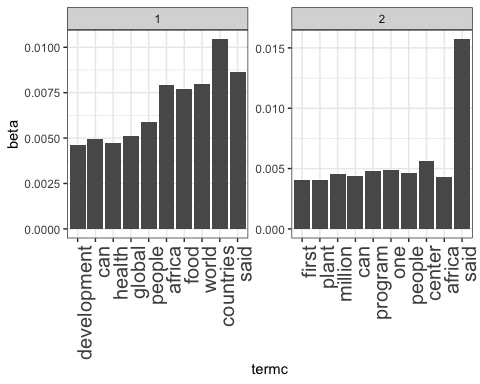
# rename the "term" column to "word"  
colnames(dtm\_tdc) <- c("document", "word", "count")  
library(textdata)  
sent <- get\_sentiments("nrc")  
nrc\_word\_countsc <- dtm\_tdc %>% inner\_join(sent, by="word") %>% count(word,sentiment, sort=TRUE) %>% ungroup()  
nrc\_word\_countsc

## # A tibble: 1,888 x 3  
## word sentiment n  
## <chr> <chr> <int>  
## 1 government fear 18  
## 2 government negative 18  
## 3 president positive 18  
## 4 president trust 18  
## 5 working positive 17  
## 6 aid positive 16  
## 7 foreign negative 15  
## 8 including positive 14  
## 9 united positive 14  
## 10 united trust 14  
## # … with 1,878 more rows

library(ggplot2)  
nrc\_word\_countsc %>%  
filter(n > 3) %>%  
mutate(n = ifelse(sentiment == "negative", -n, n)) %>%  
mutate(word = reorder(word, n)) %>%  
ggplot(aes(word, n, fill = sentiment)) +  
geom\_bar(stat = "identity") +  
theme(axis.text.x = element\_text(angle = 90, hjust = 1)) + ylab("Contribution to sentiment")



library(tidytext)  
library(topicmodels)  
dtm\_ldac <- LDA(dtmc, k = 2, control = list(seed = 1234))  
dtm\_topicsc <- tidy(dtm\_ldac, matrix = "beta")  
library(ggplot2)  
library(dplyr)  
dtm\_top\_termsc <- dtm\_topicsc %>%  
 group\_by(topic) %>%  
top\_n(10, beta) %>%  
ungroup() %>%  
arrange(topic, -beta)  
library(NLP, pos=15)  
library(tidytext, pos=18)  
library(ggplot2, pos=20)  
library(tidyr)  
beta\_spreadc <- dtm\_topicsc %>%  
mutate(topic = paste0("topic", topic)) %>%  
spread(topic, beta) %>%  
filter(topic1 > .001 | topic2 > .001) %>%  
mutate(log\_ratio = log2(topic2 / topic1))  
library(ggplot2)  
theme\_set(theme\_bw())  
dtm\_top\_termsc %>%  
mutate(termc = reorder(term, beta)) %>%  
ggplot(aes(termc, beta)) +  
geom\_bar(stat = "identity") +  
facet\_wrap(~ topic, scales = "free") +  
theme(axis.text.x = element\_text(size = 15, angle = 90, hjust = 1))

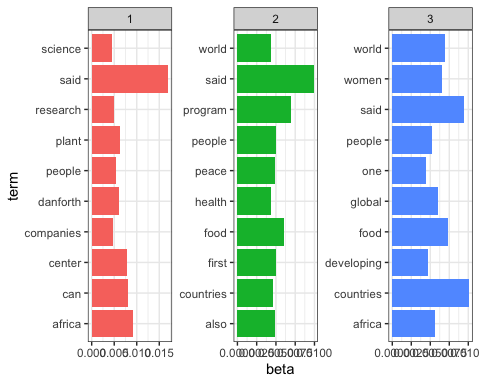


#Create and Clean Corpus  
setwd("~/Desktop/USAID Final Project")  
Combined\_corpus <- Corpus(DirSource("LoisTxfiles"))  
Combined\_corpus <- tm\_map(Combined\_corpus, removePunctuation)  
Combined\_corpus <- tm\_map(Combined\_corpus, tolower)  
Combined\_corpus <- tm\_map(Combined\_corpus, removeWords, stopwords("english"))

#Run LDA analysis to determine the topics  
Combined\_df <- DocumentTermMatrix(Combined\_corpus)  
Combined\_lda <- LDA(Combined\_df, k = 3, control = list(seed = 1234))  
Combined\_topics <- tidy(Combined\_lda, matrix = "beta")  
Combined\_top\_terms <- Combined\_topics %>% group\_by(topic) %>% top\_n(10, beta) %>%arrange(topic, -beta)  
  
#Display the topic model using ggplot  
Combined\_top\_terms %>% mutate(term = reorder(term, beta)) %>% ggplot(aes(term, beta, fill= factor(topic))) + geom\_col(show.legend = FALSE) + facet\_wrap(~topic, scales = "free") +coord\_flip()

## Warning in mutate\_impl(.data, dots, caller\_env()): Unequal factor levels:  
## coercing to character

## Warning in mutate\_impl(.data, dots, caller\_env()): binding character and  
## factor vector, coercing into character vector  
  
## Warning in mutate\_impl(.data, dots, caller\_env()): binding character and  
## factor vector, coercing into character vector  
  
## Warning in mutate\_impl(.data, dots, caller\_env()): binding character and  
## factor vector, coercing into character vector



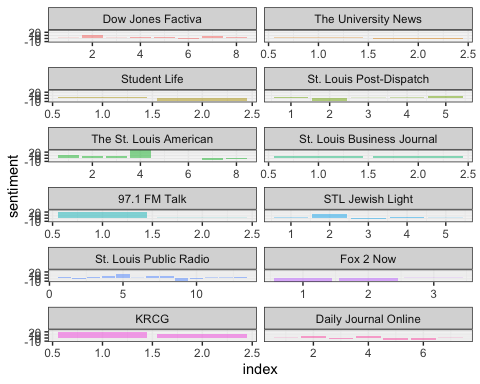
#read text from each publication articles(in a .text file)  
setwd("~/Desktop/USAID Final Project/LoisTxfiles/")  
txt1 <- readLines("1.txt")  
txt2 <- readLines("2.txt")  
txt3 <- readLines("3.txt")  
txt4 <- readLines("4.txt")  
txt5 <- readLines("5.txt")  
txt6 <- readLines("6.txt")  
txt7 <- readLines("7.txt")  
txt8 <- readLines("8.txt")  
txt9 <- readLines("9.txt")  
txt10 <- readLines("10.txt")  
txt11 <- readLines("11.txt")  
txt12 <- readLines("12.txt")  
txt13 <- readLines("13.txt")  
txt14 <- readLines("14.txt")  
txt15 <- readLines("15.txt")  
txt16 <- readLines("16.txt")  
txt17 <- readLines("17.txt")  
txt18 <- readLines("18.txt")  
txt19 <- readLines("19.txt")  
txt20 <- readLines("20.txt")  
txt21 <- readLines("21.txt")  
txt22 <- readLines("22.txt")  
txt23 <- readLines("23.txt")  
txt24 <- readLines("24.txt")  
txt25 <- readLines("25.txt")  
txt26 <- readLines("26.txt")  
txt27 <- readLines("27.txt")  
txt28 <- readLines("28.txt")  
txt29 <- readLines("29.txt")  
txt30 <- readLines("30.txt")

titles <- c("Dow Jones Factiva", "Dow Jones Factiva", "Dow Jones Factiva", "Dow Jones Factiva","The University News","Student Life","Student Life","St. Louis Post-Dispatch","St. Louis Post-Dispatch", "The St. Louis American","The St. Louis American","St. Louis Business Journal","St. Louis Business Journal","97.1 FM Talk","STL Jewish Light","St. Louis Public Radio","St. Louis Public Radio", "Fox 2 Now","KRCG","Daily Journal Online","Daily Journal Online","Daily Journal Online","St. Louis Post-Dispatch","STL Jewish Light", "STL Jewish Light", "The St. Louis American","The St. Louis American","97.1 FM Talk","St. Louis Public Radio", "St. Louis Public Radio")#, "Report News Today")#, #"NBC Washington", "NBC Washington" )  
books <- list(txt1, txt2, txt3, txt4, txt5, txt6, txt7, txt8, txt9, txt10, txt11, txt12, txt13, txt14, txt15, txt16, txt17, txt18, txt19, txt20, txt21, txt22, txt23, txt24, txt25, txt26, txt27, txt28, txt29, txt30)   
  
series <- tibble()

#Sentiment Analysis  
for( i in seq\_along(titles)) {  
 clean <- tibble(chapter = seq\_along(books[i]),  
 text = books[[i]]) %>%  
 unnest\_tokens(word, text) %>%  
 mutate(book = titles[i]) %>%  
 select(book, everything())  
 series <-rbind(series, clean)  
}  
  
series$book <- factor(series$book, levels = rev(unique(titles)))

series %>%   
 group\_by(book) %>%  
 mutate(word\_count = 1:n(),   
 index = word\_count %/% 500+ 1) %>%   
 inner\_join(get\_sentiments("bing")) %>%  
 count(book, index = index , sentiment)%>%  
 ungroup() %>% spread(sentiment, n, fill = 0) %>%  
 mutate(sentiment = positive - negative,book = factor(book, levels = unique(titles))) %>%  
 ggplot(aes(index, sentiment, fill = book)) +geom\_bar(alpha = 0.5, stat = "identity", show.legend = FALSE) + facet\_wrap(~ book, ncol = 2,scales = "free\_x")

## Joining, by = "word"



Everything below here is where the code stops working, I commented it out so that it would not run.