Overview:

In unga_tokens.rmd, I figured out an approach to n-gram tokenization that will allow for more meaningful tokenization.

```
# BASH for navigation to cloud directory

#cd /mnt/c/Users/spatt/"OneDrive - McGill University"/patterson_pouliot/inequality/inequali


# R
#setwd("C:/Users/spatt/OneDrive - McGill University/patterson_pouliot/inequality/inequality
```

```
pacman::p_load(dplyr,tokenizers,ggplot2,readr,tidytext,tidyverse,quanteda,text2vec,tidyr)
```

```
unga_tibble <- tibble(read_csv("unga_2017_corpus_unregional.csv"))
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   doc_id = col_character(),
##   text = col_character(),
##   docvar1 = col_character(),
##   docvar2 = col_double(),
##   docvar3 = col_double(),
##   P5 = col_double(),
##   NWS = col_double(),
##   ECOWAS = col_double(),
##   EU = col_double(),
##   UN_AFRICA = col_double(),
##   UN_ASIAPAC = col_double(),
##   UN_EASTEUROPE = col_double(),
##   UN_GRULAC = col_double(),
##   UN_WEOG = col_double(),
##   UN_REGION = col_character()
## )
```

```
cols <- names(unga_tibble)
cols[1] <- "index"
cols[4] <- "state"
cols[5] <- "assembly"
cols[6] <- "year"
colnames(unga_tibble) <- cols
rm(cols)

data("stop_words")
```

sixgram tokenization

First, I n-gram tokenize unga_tibble at n = 6. Since the resulting object saves the six words into a single column, I then `seperate` them into 6 individual columns and filter out stopwords from each column. The result is a dataframe with all of the non-stopworded 6-grams in the UNGA corpus, with each unit affixed to its corresponding metadata. Finally, I group the words by document and sort them by frequency. This step makes it easier to view the most frequently appearing 6-grams and it reduces the burden on RAM by dropping non-identifying metadata.

```
unga_sixgrams <- unga_tibble %>%
  unnest_tokens(sixgram, text, token = "ngrams", n = 6)

unga_sixgrams <- unga_sixgrams %>%
  separate(sixgram, c("word1","word2","word3","word4","word5","word6"), sep = " ") %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word3 %in% stop_words$word) %>%
  filter(!word4 %in% stop_words$word) %>%
  filter(!word5 %in% stop_words$word) %>%
  filter(!word6 %in% stop_words$word)

unga_sixgrams <- unga_sixgrams %>%
  group_by(doc_id) %>%
  count(word1,word2,word3,word4,word5,word6, sort = TRUE)

unga_sixgrams
```

```
## # A tibble: 35,709 x 8
## # Groups:   doc_id [6,609]
##    doc_id          word1     word2    word3    word4    word5    word6       n
##    <chr>           <chr>     <chr>    <chr>    <chr>    <chr>    <chr>   <int>
##  1 GMB_50_1995.txt 22nd      plenary  meeting  fiftieth session  13          5
##  2 GMB_50_1995.txt assembly  22nd     plenary  meeting  fiftieth session     5
##  3 GMB_50_1995.txt meeting   fiftieth session  13       october  1995        5
##  4 GMB_50_1995.txt plenary   meeting  fiftieth session  13       october     5
##  5 ZAF_50_1995.txt 22nd      plenary  meeting  fiftieth session  13          5
##  6 ZAF_50_1995.txt assembly  22nd     plenary  meeting  fiftieth session     5
##  7 ZAF_50_1995.txt meeting   fiftieth session  13       october  1995        5
##  8 ZAF_50_1995.txt plenary   meeting  fiftieth session  13       october     5
##  9 PRK_50_1995.txt beloved   leader   comrade  kim      il       sung        4
## 10 PRK_50_1995.txt respected supreme  leader   comrade  kim      jong        4
## # ... with 35,699 more rows
```

I will now isolate the semantically-coherent 6-grams. The dimensions of `unga_sixgrams` are 35709 x 8, meaning I will have to manually inspect 35709 6-grams for semantic coherence. The semantically coherent 6-grams will form

the basis of a Document-Term Matrix (DTM) that will add dimensionalty to future topic models. This process builds `human expertise` into the tokenization process.

After isolating the semantically-coherent 6-grams, they will be removed from the corpus and the process will be repeated with 5- through 1-gram models, resulting in an expert-informed DTM for further analysis.