Acled Technical Assignment - Human Rights Analysis

Load Libraries

```r
library(knitr)
library(tm)
```

```
## Loading required package: NLP
```

```r
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'

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

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.9.2     v tibble    3.2.1
## v purrr     1.0.1     v tidyr     1.3.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::annotate() masks NLP::annotate()
## x dplyr::filter()     masks stats::filter()
## x dplyr::lag()        masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(quanteda)
```

```
## Warning in .recacheSubclasses(def@className, def, env): undefined subclass
## "pcorMatrix" of class "replValueSp"; definition not updated

## Warning in .recacheSubclasses(def@className, def, env): undefined subclass
## "pcorMatrix" of class "xMatrix"; definition not updated

## Warning in .recacheSubclasses(def@className, def, env): undefined subclass
## "pcorMatrix" of class "mMatrix"; definition not updated
```

```
## Package version: 3.3.1
## Unicode version: 14.0
## ICU version: 71.1
## Parallel computing: 8 of 8 threads used.
## See https://quanteda.io for tutorials and examples.
##
## Attaching package: 'quanteda'
##
## The following object is masked from 'package:tm':
##
##     stopwords
##
## The following objects are masked from 'package:NLP':
##
##     meta, meta<-
```

```
library(readtext)
```

```
##
## Attaching package: 'readtext'
##
## The following object is masked from 'package:quanteda':
##
##     texts
```

```
library(stm)
```

```
## stm v1.3.6 successfully loaded. See ?stm for help.
##  Papers, resources, and other materials at structuraltopicmodel.com
```

```
library(tidytext)
library(ggthemes)
library(quanteda.textplots)
```

Read Data

Read data – a folder which contains a txt file for each report labelled in the format reportname.txt.

For this analysis I used a dataset from Christopher et.al published on the Harvard Dataverse. Here is the citation for the same.:

Christopher J. Fariss; Fridolin J. Linder; Zachary M. Jones; Charles D. Crabtree; Megan A. Biek; Ana-Sophia M. Ross; Taranamol Kaur; Michael Tsai, 2015, "Human Rights Texts: Converting Human Rights Primary Source Documents into Data", https://doi.org/10.7910/DVN/IAH8OY, Harvard Dataverse, V3

```
#Read Data
text <- readtext("dataverse_files_acled/dataverse/*.txt")
```

Sampling

Due to computational limitations, I take a simple random sample of 1000 texts.

```
#Sample
articles <- text %>%
  sample_n(1000)
```

Pre Processing

This data was analyzed using Quanteda and tidyverse in R. After sampling the number of documents, I began pre-processing the data. I create tokens to reduce the text into smaller, more interpretable objects. Thereafter, I perform a series of common pre-processing practices that reduce noise, increase computational efficiency, and make topic models generate topics that are concise and coherent. Some of these pre-processing steps include removing punctuation, numbers, urls and stop words common in the English language. I also create compound tokens to indicate the combination of words being in unison such as – "human rights" "u.s" etc. After performing a series of pre-processing texts, I create a document frequency matrix that describes frequency of terms in each document.

Given the limitations of processing-power, please note pre-processing is an iterative step and this would become clearer when the topics are generated in the end. Despite performing these pre-processing steps, the data often carries noise because of differing writing styles, context of the themes and topics being analyzed etc. To ideate on pre-processing further, I would like to discuss this further with any technical stakeholders and substantive experts who are well versed with literature in this field of research.

```
#select text column from articles
tokens <- articles$text %>%
  #tokenize to words
        tokens(what = "word",
               #remove punctuation
               remove_punct = TRUE,
               #remove numbers
               remove_numbers = TRUE,
               #remove urls
               remove_url = TRUE
               ) %>%
  #change all tokens to lowercase
  tokens_tolower() %>%
  #remove common stop words from the english language
  tokens_remove(stopwords("english")) %>%
  #stem using quanteda's language stemmer
  #lemmetization potential here#
  tokens_wordstem(language = quanteda_options("language_stemmer")) %>%
  #compound token to keep the word "human right" together
  #add un here
  tokens_compound(pattern = c("human right*", "u.s.*", "domestic violence*", "un*"))

#applying relative pruning, create document feature matrix where the minimum term frequency is set to 3
dfm <- dfm_trim(dfm(tokens), min_docfreq = 0.30, max_docfreq = 0.90, min_termfreq = 75, docfreq_type =
```

```
## Removing features occurring:
```

```
##   - fewer than 75 times: 68,901
```

```
##   - in fewer than 300 documents: 71,560

##   - in more than 900 documents: 11

##   Total features removed: 71,571 (98.9%).
```

```r
#remove additional characters
dfm <- dfm_remove(dfm,c("<",">", "however", "although", "$", "also"))
```

```r
textplot_wordcloud(dfm, max_words = 50, random_order = TRUE, color = "#0086b3")
```


```r
#convert dfm into a stm structure that is compatible with analysis in library(stm)
dfm_stm <- convert(dfm, to = "stm")
```

Modelling

To run a Structured Topic Model, I begin by running a search K function which enables me to test the optimal number of topics that can be generated from this text. These Ks are usually analyzed using evaluation metrics for goodness of fit like Coherence, Residuals, Lower Bound and Exclusivity. K = 7 seems to be an optimal fit for the model from an initial look at the evaluation metrics however, this is also something I would usually discuss with stakeholders or fellow technical members of the team. After running the model for K = 7, I plot the proportion of topical prevalence in the texts and the top 7 words that exist in each topic. Lastly, I also create a gamma matrix which gives me the probability of each document being associated with a topic.

```r
#Select the number of K to search optimal number of topics
K = c(5,6,7,8,9,10)

#Run Search K model to check goodness of fit for each K
model_test <- searchK(dfm_stm$documents, dfm_stm$vocab, K = K, verbose = TRUE)
```

```r
# Plot Eval Metrics for checking model fit
plot <- data.frame("K" = K,
                   "Coherence" = unlist(model_test$results$semcoh),
                   "Exclusivity" = unlist(model_test$results$exclus),
                   "Residual" =  unlist(model_test$results$residual),
                   "Lower Bound" = unlist(model_test$results$lbound))

# Reshape to long format
library("reshape2")
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

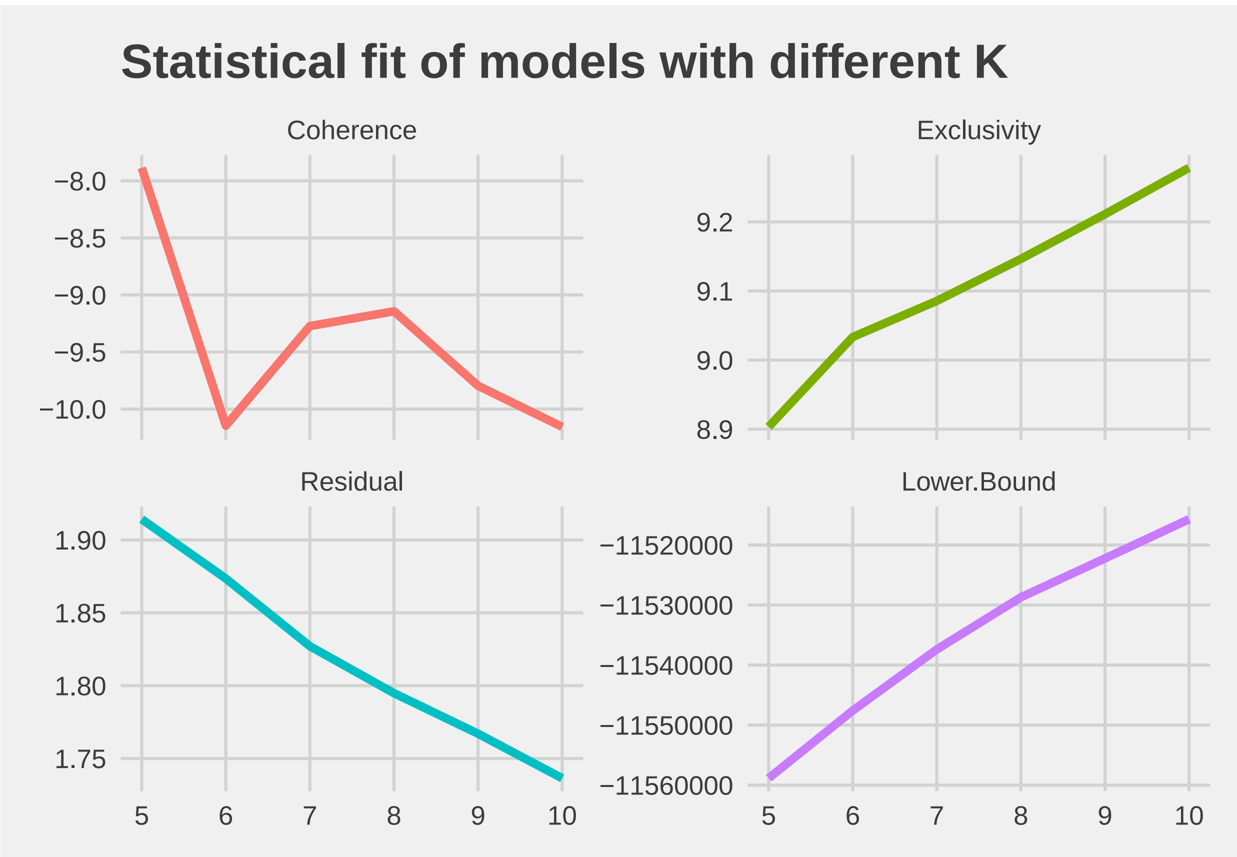
```r
plot <- melt(plot, id=c("K"))
plot
```

```
##     K    variable         value
## 1   5   Coherence -7.885670e+00
## 2   6   Coherence -1.014604e+01
## 3   7   Coherence -9.272952e+00
## 4   8   Coherence -9.143177e+00
## 5   9   Coherence -9.796243e+00
## 6  10   Coherence -1.015612e+01
## 7   5 Exclusivity  8.903410e+00
## 8   6 Exclusivity  9.033407e+00
## 9   7 Exclusivity  9.085631e+00
## 10  8 Exclusivity  9.146418e+00
## 11  9 Exclusivity  9.211176e+00
## 12 10 Exclusivity  9.278265e+00
## 13  5    Residual  1.914245e+00
## 14  6    Residual  1.873528e+00
## 15  7    Residual  1.827059e+00
## 16  8    Residual  1.794833e+00
## 17  9    Residual  1.767018e+00
## 18 10    Residual  1.736443e+00
## 19  5 Lower.Bound -1.155886e+07
## 20  6 Lower.Bound -1.154759e+07
## 21  7 Lower.Bound -1.153739e+07
## 22  8 Lower.Bound -1.152871e+07
## 23  9 Lower.Bound -1.152223e+07
## 24 10 Lower.Bound -1.151570e+07
```

```r
library("ggplot2")
fit_stats <- ggplot(plot, aes(K, value, color = variable)) +
  geom_line(size = 1.5, show.legend = FALSE) +
  facet_wrap(~variable,scales = "free_y") +
  labs(x = "Number of topics K",
       title = "Statistical fit of models with different K")+
  theme_fivethirtyeight()
```

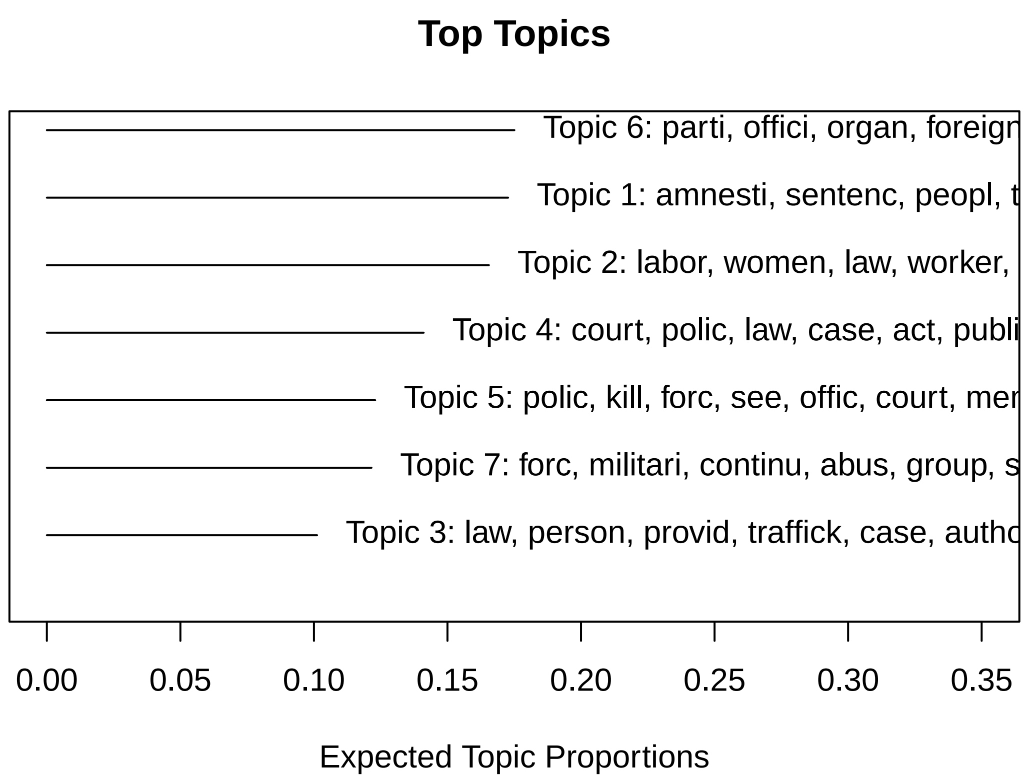
```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
print(fit_stats)
```




```r
# Run the model with K = best fit scores
model <- stm(documents = dfm_stm$documents,
          vocab = dfm_stm$vocab,
          K = 7,
          verbose = TRUE)
```

```r
#plot the topics and the top 7 words from the topic
plot.STM(model, "summary", n=7)
```




```r
# For each topic, print the first seven common words, use FREX score to evaluate model
print(labelTopics(model,topics = c(1:7), n=7))
```

```
## Topic 1 Top Words:
##       Highest Prob: amnesti, sentenc, peopl, trial, releas, death, tortur
##       FREX: amnesti, peopl, sentenc, releas, trial, execut, death
##       Lift: amnesti, appar, imprison, peopl, execut, sentenc, un
##       Score: amnesti, peopl, death, sentenc, imprison, execut, un
## Topic 2 Top Words:
##       Highest Prob: labor, women, law, worker, provid, person, howev
##       FREX: labor, employ, worker, sector, women, wage, age
##       Lift: workweek, sector, bargain, wage, minimum, compulsori, femal
##       Score: workweek, labor, percent, employ, child, disabl, section
## Topic 3 Top Words:
##       Highest Prob: law, person, provid, traffick, case, author, polic
##       FREX: traffick, sexual, corrupt, victim, ministri, person, access
##       Lift: perform, traffick, sexual, corrupt, center, fine, registr
##       Score: perform, traffick, disabl, percent, sexual, child, law
## Topic 4 Top Words:
##       Highest Prob: court, polic, law, case, act, public, minist
##       FREX: penalti, act, minist, parliament, rule, express, legisl
##       Lift: penalti, amend, propos, recommend, prime, review, challeng
```

```
##       Score: penalti, polic, court, act, law, death, parliament
## Topic 5 Top Words:
##       Highest Prob: polic, kill, forc, see, offic, court, member
##       FREX: kill, see, polic, offic, beat, station, forc
##       Lift: inquiri, shot, injur, kill, beat, see, station
##       Score: inquiri, polic, kill, see, forc, shot, section
## Topic 6 Top Words:
##       Highest Prob: parti, offici, organ, foreign, public, freedom, author
##       FREX: religi, foreign, parti, travel, permit, religion, control
##       Lift: membership, abroad, permiss, emigr, travel, guarante, correspond
##       Score: membership, percent, religion, emigr, foreign, religi, freedom
## Topic 7 Top Words:
##       Highest Prob: forc, militari, continu, abus, group, secur, civilian
##       FREX: war, unit, civilian, displac, attack, effort, aid
##       Lift: displac, war, humanitarian, conflict, thousand, aid, armi
##       Score: displac, war, armi, soldier, militari, attack, civilian
```

```r
#Save top 20 features across topics and forms of weighting
labels <- labelTopics(model, n=30)
#only keep FREX weighting
topwords <- data.frame("features" = t(labels$frex))
#assign topic number as column name
colnames(topwords) <- paste("Topics", c(1:7))
#Return the result
print(topwords[1:7])
```

```
##     Topics 1  Topics 2  Topics 3      Topics 4    Topics 5   Topics 6
## 1    amnesti     labor  traffick       penalti        kill     religi
## 2      peopl    employ    sexual           act         see    foreign
## 3    sentenc    worker   corrupt        minist       polic      parti
## 4     releas    sector    victim    parliament       offic     travel
## 5      trial     women  ministri          rule        beat     permit
## 6     execut      wage    person       express     station   religion
## 7      death       age    access        legisl        forc    control
## 8   imprison  prohibit    provid       concern  journalist     econom
## 9       held   minimum    center         prime        shot     church
## 10    tortur   practic       law       commiss     suspect    citizen
## 11    detain constitut  children         order      presid    freedom
## 12   detaine   respect     child       council    demonstr      activ
## 13        un   bargain     minor         amend         end      press
## 14    detent    provid     media        review      result   restrict
## 15     least discrimin   approxim     recommend      attack      grant
## 16 disappear    privat    disabl         court       feder    opposit
## 17    appeal   section    accord          issu     section    permiss
## 18    receiv   industri prosecutor     prosecut       often      power
## 19     alleg     child    enforc         decis      leader membership
## 20     appar    strike   problem        crimin     inquiri      exist
## 21   convict     union   individu          face    district   independ
## 22   without    domest   percent           new presidenti    assembl
## 23     other     femal  discrimin        propos    militari   movement
## 24      said   percent     facil       convent      member     abroad
## 25     charg  children    domest     complaint        fire     social
## 26       tri    employe      lack        reform        land     polici
## 27     three      hour    inform parliamentari      injur   particip
```

```
## 28    former   general    protect        depart     august     associ
## 29    beaten      free     servic        immigr     suprem     recent
## 30     known    tradit     regist         crime      local     critic
##         Topics 7
## 1            war
## 2           unit
## 3       civilian
## 4        displac
## 5         attack
## 6         effort
## 7            aid
## 8        monitor
## 9            arm
## 10      conflict
## 11          abus
## 12       million
## 13     administr
## 14          armi
## 15       develop
## 16   humanitarian
## 17      militari
## 18       support
## 19       soldier
## 20          camp
## 21        assist
## 22      thousand
## 23          area
## 24          peac
## 25        region
## 26        violat
## 27        refuge
## 28     agreement
## 29        return
## 30          late
```

```r
#probability of each document being associated with each topic (Sample head(10))
theta <- make.dt(model)
theta[1:10,1:8]
```

```
##     docnum      Topic1       Topic2      Topic3     Topic4     Topic5
##  1:      1 0.001855896 0.2060411555 0.460577507 0.04149653 0.12319775
##  2:      2 0.048943089 0.0894189855 0.077369210 0.02778784 0.02774484
##  3:      3 0.688578045 0.0082267567 0.009792720 0.05115274 0.08138651
##  4:      4 0.021043309 0.0110439321 0.075584421 0.05210252 0.25484251
##  5:      5 0.008645957 0.2757210686 0.454137463 0.07864648 0.15139051
##  6:      6 0.030079099 0.2985596687 0.009769224 0.07146722 0.08650106
##  7:      7 0.007060351 0.2926823315 0.462785540 0.02849059 0.12505874
##  8:      8 0.332470535 0.0008212083 0.008207904 0.17247253 0.32708195
##  9:      9 0.044846869 0.2909540748 0.331231901 0.03167190 0.16267729
## 10:     10 0.040872475 0.0147175904 0.114855578 0.06497320 0.36450167
##          Topic6     Topic7
##  1: 0.074327698 0.09250346
##  2: 0.716862017 0.01187402
##  3: 0.147144887 0.01371834
```

```
##  4: 0.216728636 0.36865467
##  5: 0.006995748 0.02446277
##  6: 0.460867724 0.04275600
##  7: 0.017545124 0.06637732
##  8: 0.006742172 0.15220370
##  9: 0.057486383 0.08113158
## 10: 0.016722928 0.38335657
```

Visualization and Insights (Part 2 of the Technical Assignment)

In this section, I develop some exploratory graphs to see the topic prevalance in our sample. Fig 1.1 shows the proportion of topical prevalance in our sample, and I overlay the top 7 words the occur in each topic. Fig 1.2 goes a step further, and gives the proportion of the words occurring in each topic. These two graphs serve as an inital exploration point to observe keywords and see if there are thematic trends prevelant in the data. In addition, I also create a time-series which show the number of texts published by each organization in each year. Here n = 14190, which is the entirety of the dataset.

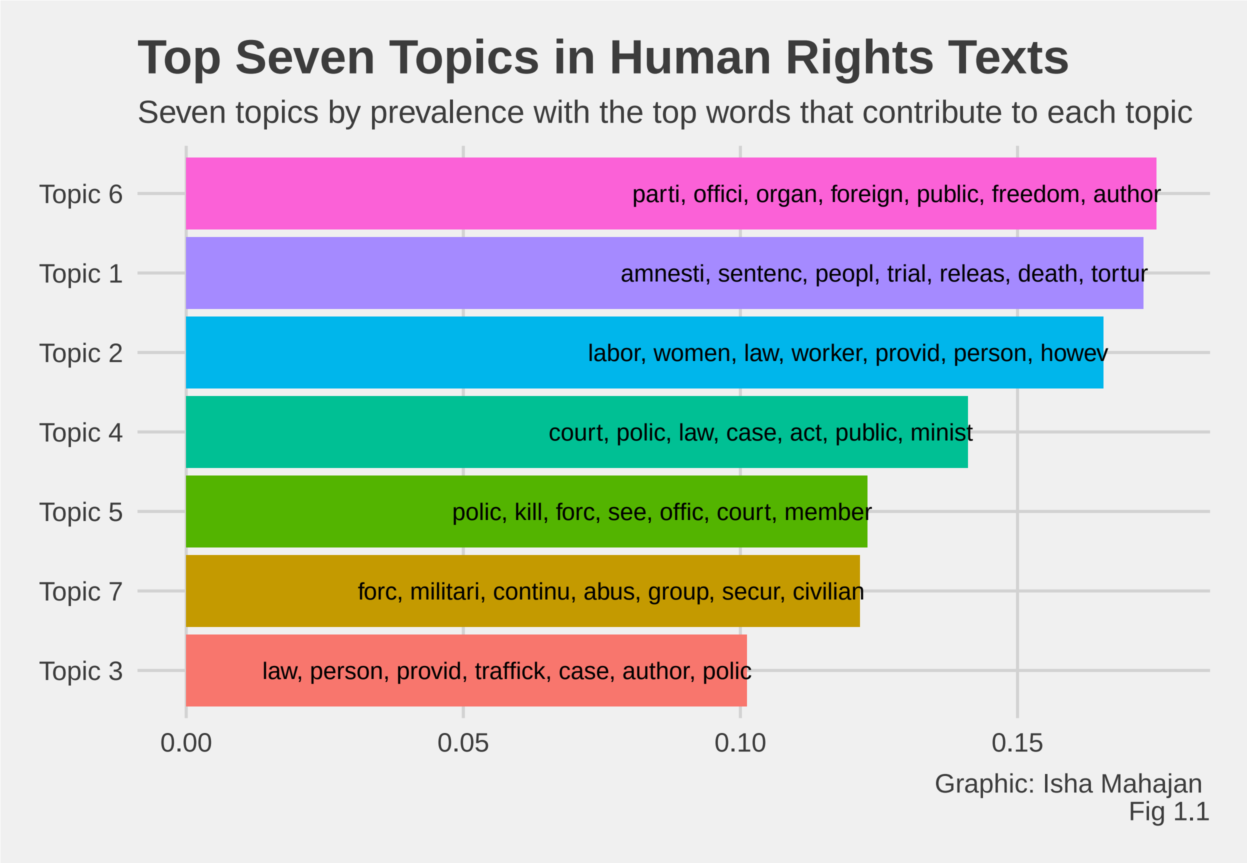
```r
#convert model into tidy tibble
model_beta <- tidy(model)
head(model_beta)
```

```
## # A tibble: 6 x 3
##   topic term       beta
##   <int> <chr>     <dbl>
## 1     1 abl   0.000224
## 2     2 abl   0.000325
## 3     3 abl   0.000204
## 4     4 abl   0.0000638
## 5     5 abl   0.000132
## 6     6 abl   0.00109
```

```r
#covert model into tibble gamma matrix -- probability of each document being associated with a topic
model_gamma <- tidy(model, matrix = "gamma",
                 document_names = rownames(articles))
model_gamma
```

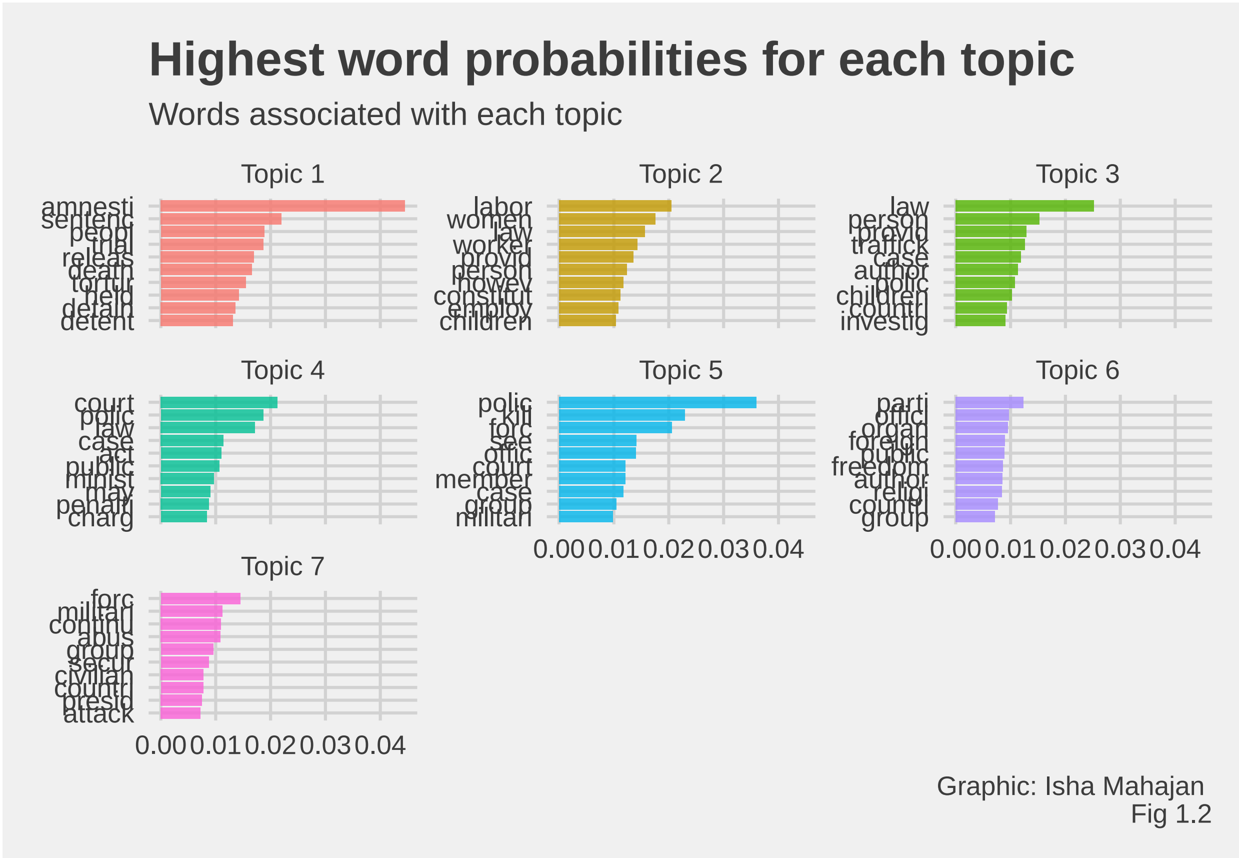
```
## # A tibble: 7,000 x 3
##    document topic   gamma
##    <chr>    <int>   <dbl>
##  1 1            1 0.00186
##  2 2            1 0.0489
##  3 3            1 0.689
##  4 4            1 0.0210
##  5 5            1 0.00865
##  6 6            1 0.0301
##  7 7            1 0.00706
##  8 8            1 0.332
##  9 9            1 0.0448
## 10 10           1 0.0409
## # i 6,990 more rows
```

```r
top_terms <- model_beta%>%
  arrange(beta) %>%
  group_by(topic) %>%
  top_n(7, beta) %>%
  arrange(-beta) %>%
  select(topic, term) %>%
  summarise(terms = list(term)) %>%
  mutate(terms = map(terms, paste, collapse = ", ")) %>%
  unnest(cols = c(terms))

gamma_terms <- model_gamma %>%
  group_by(topic) %>%
  summarise(gamma = mean(gamma)) %>%
  arrange(desc(gamma)) %>%
  left_join(top_terms, by = "topic") %>%
  mutate(topic = paste0("Topic ", topic),
         topic = reorder(topic, gamma))

figone_one <- gamma_terms %>%
  top_n(8, gamma) %>%
  ggplot(aes(topic, gamma, label = terms, fill = topic)) +
  geom_col(show.legend = FALSE) +
  geom_text(hjust = 1, nudge_y = 0.0009, size = 3) +
  coord_flip() +
  theme_hc() +
  theme(plot.title = element_text(size = 12)) +
  labs(x = NULL, y = expression(gamma),
       title = "Top Seven Topics in Human Rights Texts",
       subtitle = "Seven topics by prevalence with the top words that contribute to each topic",
       caption = "Graphic: Isha Mahajan \nFig 1.1")+
  theme_fivethirtyeight()
print(figone_one)
```




```
#ggsave(figone_one, "fig1.1.png", dpi = 400)
```

```
figeone_two <- td_beta <- tidytext::tidy(model)
td_beta %>%
  group_by(topic) %>%
  top_n(10, beta) %>%
 ungroup() %>%
    mutate(topic = paste0("Topic ", topic),
         term = reorder_within(term, beta, topic)) %>%
  ggplot(aes(term, beta, fill = as.factor(topic))) +
  geom_col(alpha = 0.8, show.legend = FALSE) +
  facet_wrap(~ topic, scales = "free_y") +
  coord_flip() +
  scale_x_reordered() +
  labs(x = NULL, y = expression(beta),
       title = "Highest word probabilities for each topic",
       subtitle = "Words associated with each topic",
       caption = "Graphic: Isha Mahajan \n Fig 1.2")+
       scale_color_manual(aesthetics = "Darjeeling2")+
  theme_fivethirtyeight()
```




```r
print(figeone_two)
```

```
## # A tibble: 5,453 x 3
##    topic term        beta
##    <int> <chr>      <dbl>
##  1     1 abl    0.000224
##  2     2 abl    0.000325
##  3     3 abl    0.000204
##  4     4 abl    0.0000638
##  5     5 abl    0.000132
##  6     6 abl    0.00109
##  7     7 abl    0.000534
##  8     1 abroad 0.000502
##  9     2 abroad 0.000182
## 10     3 abroad 0.000142
## # i 5,443 more rows
```

```r
#ggsave(figeone_two, "fig1.2.png", dpi = 400)
```

```r
metadata <- read_csv("dataverse_files_acled/reports_metadata.csv")
```

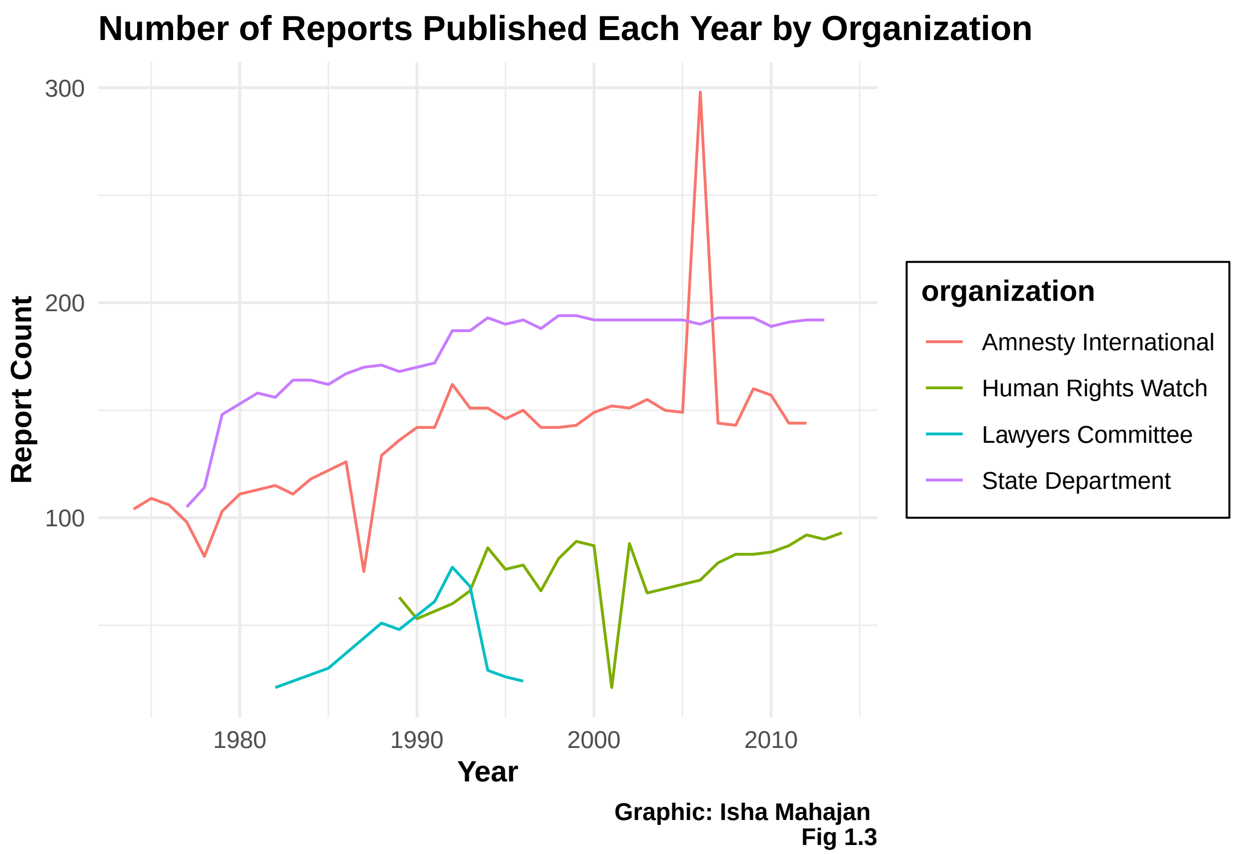
```
## Rows: 14190 Columns: 26
## -- Column specification --------------------------------------------------------
## Delimiter: ","
```

```
## chr  (6): file_name, new_filename, country_iso3c, country_name, report_name,...
## dbl (20): year.0, word_count, hathaway, state, fariss.mean, fariss.std_devia...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
metadata_transformed <- metadata %>%
  group_by(organization, year.0) %>%
  summarise(count = n())
```

```
## `summarise()` has grouped output by 'organization'. You can override using the
## `.groups` argument.
```

```r
figone_three <- ggplot(metadata_transformed, aes(x = year.0, y = count, color = organization)) +
  geom_line() +
  labs(title = "Number of Reports Published Each Year by Organization",
       x = "Year",
       y = "Report Count",
       legend = "Organization",
       caption = "Graphic: Isha Mahajan \n Fig 1.3") +
  theme_minimal() +
theme(
    title = element_text(face = "bold"),
    legend.title = element_text(face = "bold"),
    legend.box.background = element_rect(color = "black", linetype = "solid")
  )
print(figone_three)
```




```
#ggsave(figone_three, "Fig1.3.png", dpi = 400)
```

Refelections (Part 3)

This is an initial analysis and exploration to build a structural topic model and explore the potential of using Natural Language Processing in the field of Human Rights. By looking at a random sample of 1000 documents, this model, with a short run time, was able to generate topics and probabilities of a document belonging to a certain topic. This can work in parallel with human coders who have to go through volumes of texts and generate thematic codes to classify them into categories. If iterated upon, a model like this can serve as a good starting point to automate some of those processes, and serve useful to organizations like ACLED to diversify their data sources by analyzing large volumes of texts and generating insights for the broader research community in political violence and global affairs.

Keeping this model at the core of building out a process, I would like to work with this at scale depending on the computing power available. By using popular libraries in R/Python like Beautiful Soup or API calls, we could leverage large volumes of text to train a model and generate initial topical insights. Thereafter,the model can serve as a starting point to share topical prevalence of documents from websites like amnesty, human rights, landmine monitor etc. to provide the research community an opportunity to make their search processes more streamlined, enable coders to work in tandem with the model to increase it's accuracy, and eventually scale this into a predictive model where we can predict the time when the conflict would we reported, classify the organization by which a text was published etc.

The key features of this tool would be:

Generating and Contextualizing topics from large volumes of text

Opportunity to select organizations who's text the user is interested in exploring; the ability to see the topical prevalence in those text

Forecast whether a future report/document by these organizations would be classified into a certain topic or not.

Forecast the time when a conflict would be reported and perhaps exploring the lag from time of conflict to time of reporting