Word Relationships

Shale

5/2/2022

Import Data

```
files <- list.files(path = here("data"), pattern = "^EPA")

#ej_reports <- lapply(files, pdf_text)

ej_pdf <- readtext(file = here("data", "EPAEJ*"),
                   docvarsfrom = "filenames",
                   docvarnames = c("type", "year"),
                   sep = "_")

#creating an initial corpus containing our data
epa_corp <- corpus(x = ej_pdf, text_field = "text" )
summary(epa_corp)
```

```
## Corpus consisting of 6 documents, showing 6 documents:
##
##             Text Types Tokens Sentences  type year
##   EPAEJ_2015.pdf  2136   8944       263 EPAEJ 2015
##   EPAEJ_2016.pdf  1599   7965       176 EPAEJ 2016
##   EPAEJ_2017.pdf  3973  30564       653 EPAEJ 2017
##   EPAEJ_2018.pdf  2774  16658       447 EPAEJ 2018
##   EPAEJ_2019.pdf  3773  22648       672 EPAEJ 2019
##   EPAEJ_2020.pdf  4493  30523       987 EPAEJ 2020
```

Cleaning Data

```
# Adding some additional, context-specific stop words to stop word lexicon
more_stops <- c("2015","2016", "2017", "2018", "2019", "2020", "www.epa.gov", "https")
add_stops <- tibble(word = c(stop_words$word, more_stops))
stop_vec <- as_vector(add_stops)
```

```
# tokenization and cleaning
tokens <- tokens(epa_corp, remove_punct = TRUE)

toks1 <- tokens_select(tokens, min_nchar = 3) %>%
  tokens_tolower() %>%
  tokens_remove(pattern = (stop_vec))

dfm <- dfm(toks1)
```

Relationship Analysis

```r
# bigrams
toks2 <- tokens_ngrams(toks1, n=2)
dfm2 <- dfm(toks2)
dfm2 <- dfm_remove(dfm2, pattern = c(stop_vec))
freq_words2 <- textstat_frequency(dfm2, n=20)
freq_words2$token <- rep("bigram", 20)
```

```r
# trigrams
toks3 <- tokens_ngrams(toks1, n=3)
dfm3 <- dfm(toks3)
dfm3 <- dfm_remove(dfm3, pattern = c(stop_vec))
freq_words3 <- textstat_frequency(dfm3, n=20)
freq_words3$token <- rep("trigram", 20)
```

```r
freq_words2
```

```
##                      feature frequency rank docfreq group  token
## 1     environmental_justice       556    1       6   all bigram
## 2      technical_assistance       139    2       6   all bigram
## 3            drinking_water       133    3       6   all bigram
## 4             public_health       123    4       6   all bigram
## 5           progress_report       108    5       6   all bigram
## 6               air_quality        73    6       6   all bigram
## 7             water_systems        66    7       6   all bigram
## 8    vulnerable_communities        65    8       6   all bigram
## 9                epa_region        62    9       5   all bigram
## 10     environmental_public        57   10       6   all bigram
## 11         federal_agencies        56   11       6   all bigram
## 12   national_environmental        51   12       6   all bigram
## 13           justice_fy2017        51   12       1   all bigram
## 14          fy2017_progress        51   12       1   all bigram
## 15          superfund_sites        48   15       4   all bigram
## 16        indigenous_peoples        46   16       6   all bigram
## 17             civil_rights        46   16       5   all bigram
## 18        local_governments        45   18       6   all bigram
## 19             urban_waters        44   19       6   all bigram
## 20 overburdened_communities        43   20       6   all bigram
```

```r
freq_words3
```

```
##                              feature frequency rank docfreq group   token
## 1            justice_fy2017_progress        51    1       1   all trigram
## 2             fy2017_progress_report        51    1       1   all trigram
## 3          environmental_public_health        50    3       6   all trigram
## 4         environmental_justice_fy2017        50    3       1   all trigram
## 5       national_environmental_justice        37    5       6   all trigram
## 6         office_environmental_justice        32    6       6   all trigram
## 7          epa's_environmental_justice        32    6       6   all trigram
## 8       environmental_justice_progress        30    8       4   all trigram
## 9            justice_progress_report        30    8       4   all trigram
## 10      environmental_justice_concerns        30    8       5   all trigram
## 11             drinking_water_systems        29   11       5   all trigram
## 12        annual_environmental_justice        27   12       5   all trigram
## 13      environmental_justice_advisory        27   12       6   all trigram
```

```
## 14       fiscal_annual_environmental        25   14        3   all trigram
## 15          justice_advisory_council        24   15        6   all trigram
## 16      environmental_justice_grants        22   16        5   all trigram
## 17  technical_assistance_communities        20   17        6   all trigram
## 18 communities_environmental_justice        20   17        5   all trigram
## 19               safe_drinking_water        19   19        5   all trigram
## 20     technical_assistance_services        19   19        5   all trigram
```

The two tables above show the most common bigrams and trigrams in the EPAEJ Reports. A comparison of most common bigrams and trigrams shows that bigrams are probably the more useful set to look at: trigrams tend to include more "noise" (things like fy2017/office/report) but don't add any additional meaningful relationships beyond what is provided by the bigrams.

Correlation Network

```
#convert to tidy format and apply my stop words
raw_text <- tidy(epa_corp)

#Distribution of most frequent words across documents
raw_words <- raw_text %>%
  mutate(year = as.factor(year)) %>%
  unnest_tokens(word, text) %>%
  anti_join(add_stops, by = 'word') %>%
  count(year, word, sort = TRUE)

#number of total words by document
total_words <- raw_words %>%
  group_by(year) %>%
  summarize(total = sum(n))

report_words <- left_join(raw_words, total_words)
```

```
## Joining, by = "year"
```

```
par_tokens <- unnest_tokens(raw_text, output = paragraphs, input = text, token = "paragraphs")

par_tokens <- par_tokens %>%
 mutate(par_id = 1:n())

par_words <- unnest_tokens(par_tokens, output = word, input = paragraphs, token = "words") %>%
  anti_join(add_stops, by = 'word')
```

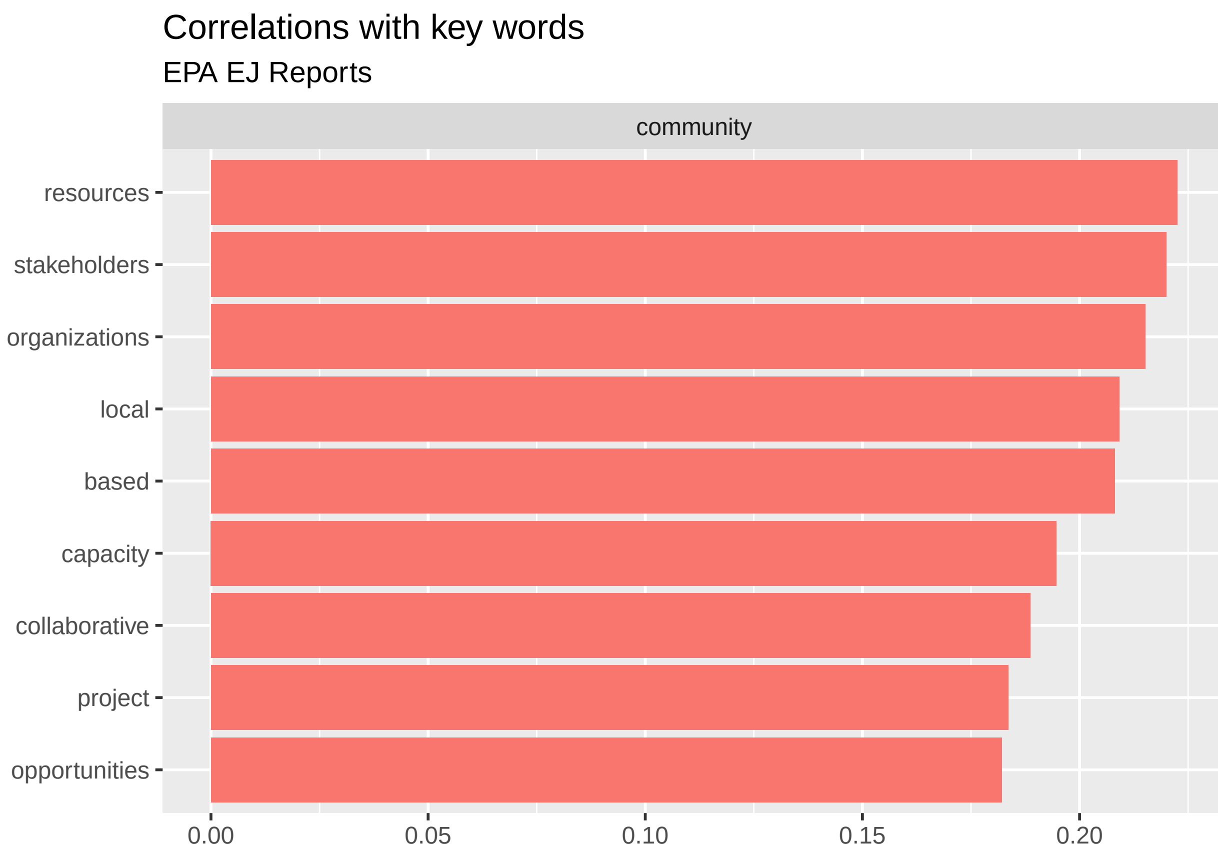
```
word_cors <- par_words %>%
  add_count(par_id) %>%
  filter(n >= 50) %>%
  select(-n) %>%
  pairwise_cor(word, par_id, sort = TRUE)

comm_cors <- word_cors %>%
  filter(item1 == "community")
```

Chart:

```
comm_cors %>%
  top_n(9) %>%
  ungroup() %>%
```

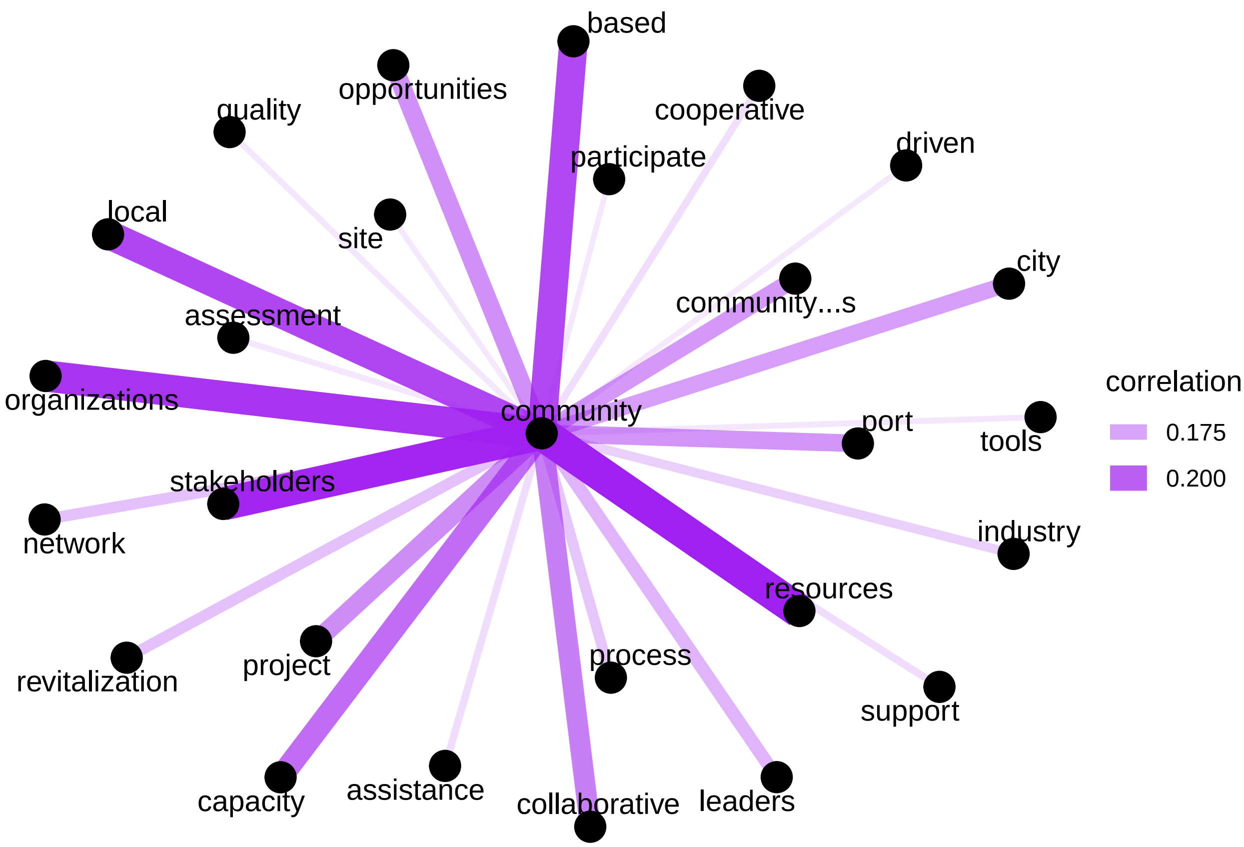
```
  mutate(item1 = as.factor(item1),
  name = reorder_within(item2, correlation, item1)) %>%
  ggplot(aes(y = name, x = correlation, fill = item1)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~item1, ncol = 2, scales = "free")+
  scale_y_reordered() +
  labs(y = NULL,
         x = NULL,
         title = "Correlations with key words",
         subtitle = "EPA EJ Reports")
```

```
## Selecting by correlation
```



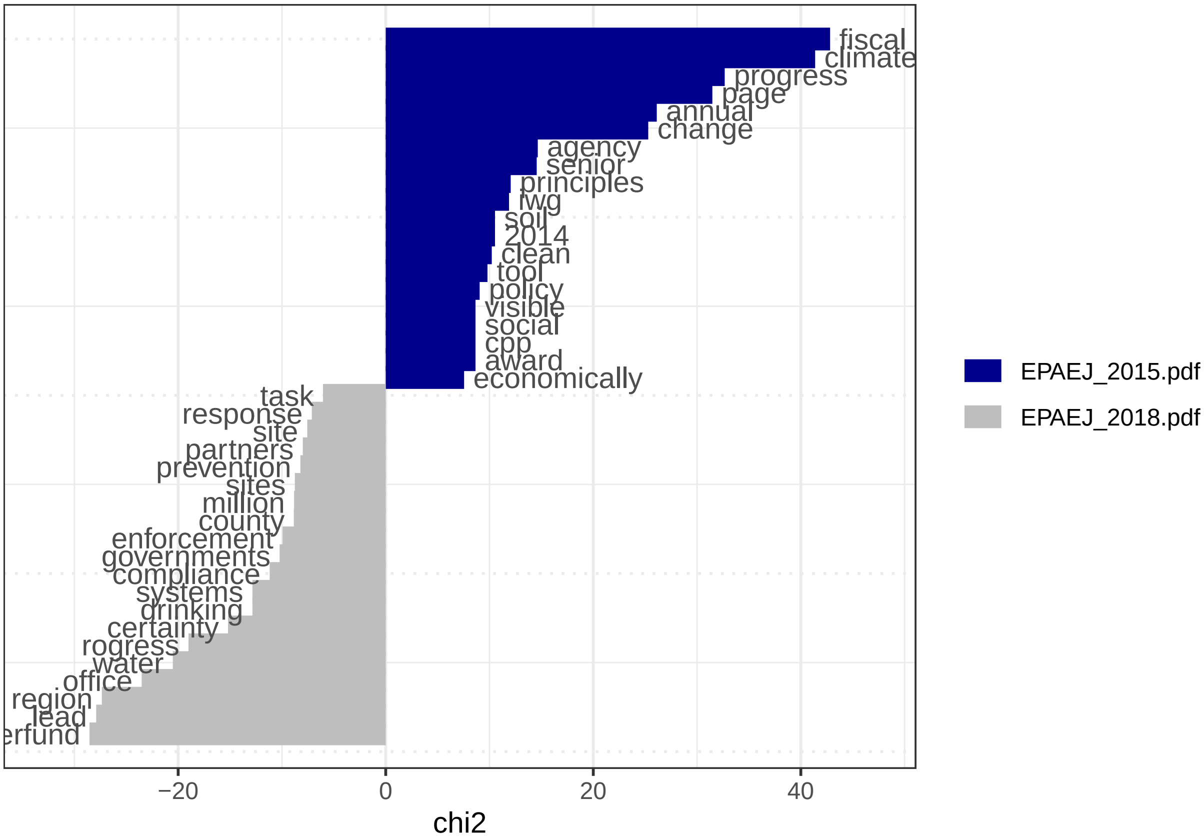
Network Visualization:

```
comm_cors  %>%
  filter(correlation > .15) %>%
  graph_from_data_frame() %>%
  ggraph(layout = "fr") +
  geom_edge_link(aes(edge_alpha = correlation, edge_width = correlation), edge_colour = "purple") +
  geom_node_point(size = 5) +
  geom_node_text(aes(label = name), repel = TRUE,
                 point.padding = unit(0.2, "lines")) +
  theme_void()
```

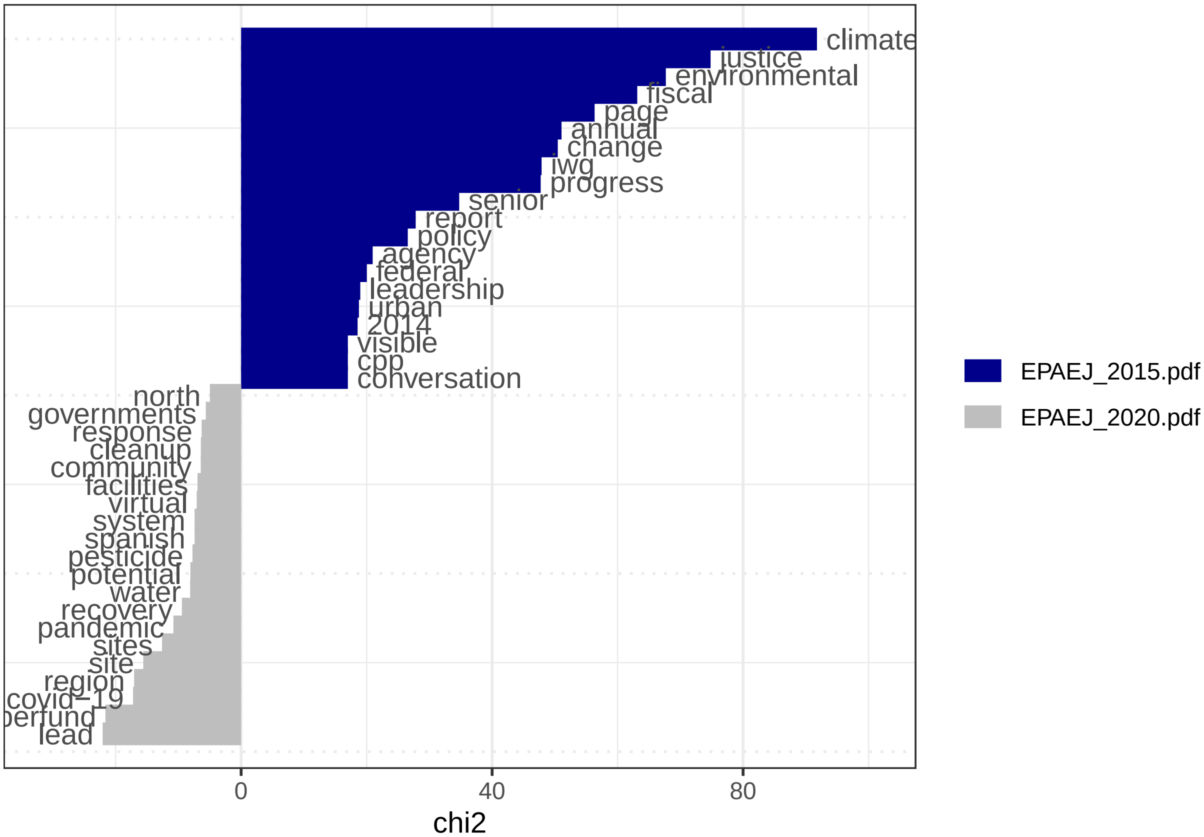


Keyness Analysis Function

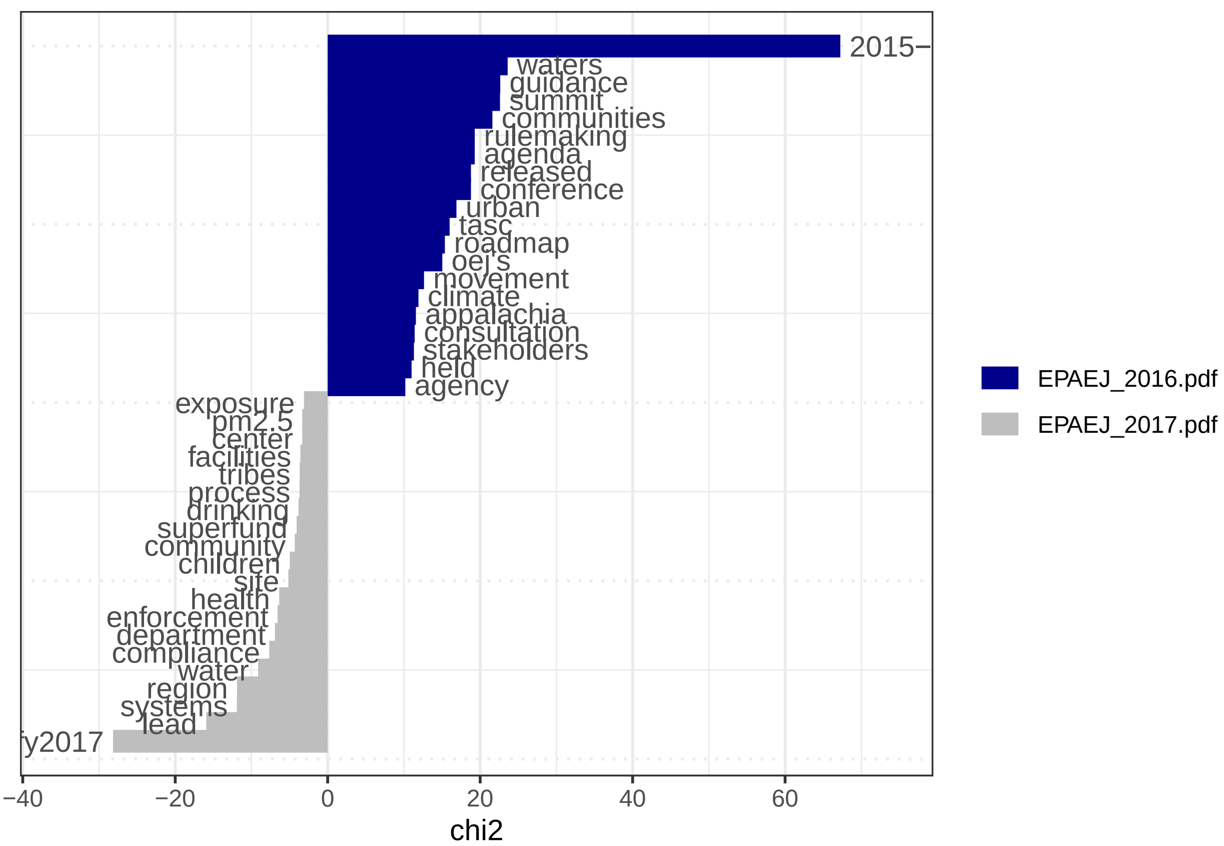
```
key_gram <- function(r1, r2) {
  docs <- dfm_subset(dfm, subset = (dfm@docvars[["docname_"]] %in% c(r1, r2)))

  keyness <- textstat_keyness(docs, target = r1)
  textplot_keyness(keyness)
}

key_gram("EPAEJ_2015.pdf", "EPAEJ_2018.pdf")
```




```
key_gram("EPAEJ_2015.pdf", "EPAEJ_2020.pdf")
```

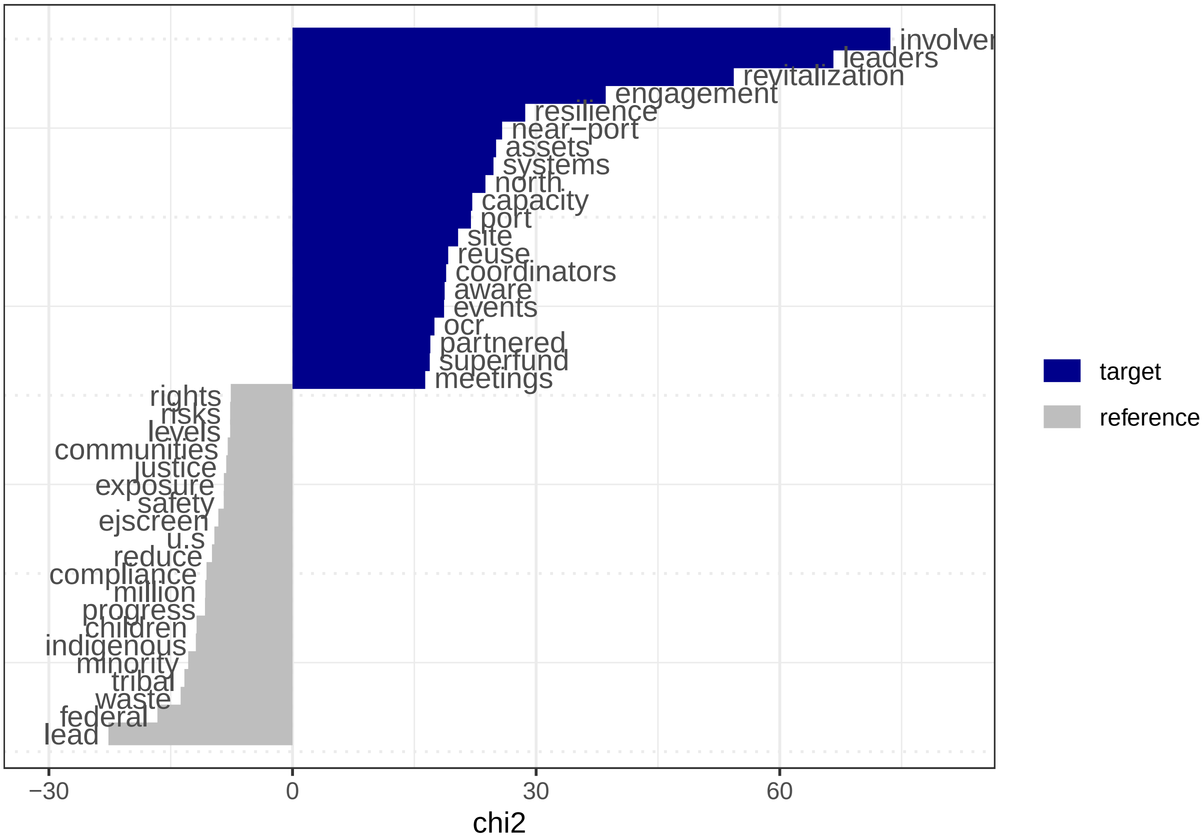



```
key_gram("EPAEJ_2016.pdf", "EPAEJ_2017.pdf")
```



10-Word Window

```
toks10 <- tokens_keep(toks1, window = 10, pattern = "community")
toks10 <- tokens_remove(toks10, pattern = "community")
tok_un <- tokens_remove(toks1, window = 10, pattern = "community")

dfm10 <- dfm(toks10)
dfm_out <- dfm(tok_un)

dfm10 <- dfm_remove(dfm10, pattern = c(stop_vec))

dfm_compare <- rbind(dfm10, dfm_out)

comm_key <- textstat_keyness(dfm_compare, target = seq_len(ndoc(dfm10)))

textplot_keyness(comm_key)
```



As indicated in the legend, the target is words within the 10-word window of "community", while the reference is all other words.