The Bag of Words

POS6933: Computational Social Science

Jake S. Truscott, Ph.D

University of Florida

Spring 2026

Overview

- Discussion Re: Topic Selection Assignment & Final Project Assessment
- Week 4 Problem Set Review
- Contextualizing the *Bag of Words* (BoW)
- Document Frequency Matrices
- Word Clouds

Topic Selection Assignment

- If haven't already: Respond to my questions/comments on Canvas
- **Big Items**:
  - Substance v. Application
  - Depth $>$ Breadth
  - Cool Data & Method $\neq$ Sufficient – Needs to have (*coherent*) structure of academic research article
  - I am a resource – collaborate and ***Don't Procrastinate***

Submission, Presentation, and Evaluation

Formatting

- Final papers must be compiled in `Latex` or `RMarkdown` and submitted as PDF.
- Must include supplemental appendix with any and all `R` or `Python` code used to render tables/figures
- Presentation slides can be in `PowerPoint` – though I can provide my template for UF `Beamer (LaTex)` for anyone interested.
- Presentations should be 12-15 minutes and allow for 5-10 minutes of Q&A (lead by intructor & peer evaluators)

Evaluation

- Topic Assignment (5pts)
- Instructor Evaluation (35pts) – *Rubric Coming Soon*
- Peer Review (5pts) – *Assignment(s) Coming Soon*

Looking Forward

- I want drafts for review by **April 1** – *I will tell you what to fix...*
- I'm happy to collaborate but not workshop – *No half-baked ideas...*
- **Seriously**... don't put this off until April – **Fair Warning**: *undeveloped work will be treated as such.*
- **Any questions re: formatting, expectations, etc.?**

Week 4 Problem Set

Notes:

- Generally good work
- I am going to start being more critical of `RMarkdown` submissions
- Most Common Problem: Text pre-processing (*More Today...*)

The Bag of Words

- **The Bag of Words**: Represents documents as a collection of individual words, ignoring grammar and word order – emphasizes co-occurrence of these terms as a principal indicator of similarity or cohesion across documents.
- Why it's Useful: Converts text into numerical features that can be used for classification, regression, and clustering tasks (*Coming Soon!*).
- In short: We'll build a *vocabulary* of all unique words across documents, then represent each document as a vector of word counts (*frequencies*) corresponding to that vocabulary.
  - Can use these vectors to inform of us both individual documents (ex: emphasize of certain words over others), as well as in comparison to other documents (e.g., how some documents use certain words more than others)

State of the Union Address

- We want to know how (if) presidents talk about `military` issues during annual State of the Union addresses.

State of the Union Address

- We want to know how (if) presidents talk about `military` issues during annual State of the Union addresses.
- **First Step**: Construct a vector of terms we associate with the topic of interest (e.g., `troops`, `defense`, `war`, `security`, `veterans`).

State of the Union Address

- We want to know how (if) presidents talk about `military` issues during annual State of the Union addresses.
- **First Step**: Construct a vector of terms we associate with the topic of interest (e.g., `troops`, `defense`, `war`, `security`, `veterans`).
- **Operationalization**: For each address, count the frequency (or proportion) of these `military`-related terms.

State of the Union Address

- We want to know how (if) presidents talk about `military` issues during annual State of the Union addresses.
- **First Step**: Construct a vector of terms we associate with the topic of interest (e.g., `troops`, `defense`, `war`, `security`, `veterans`).
- **Operationalization**: For each address, count the frequency (or proportion) of these `military`-related terms.
- **Key Assumption**: We are confident about capturing a generalizable series of statements concerning `military` issues because these specific terms should appear in any `military`-related section of the speech.

SOTU Address – Military (Cont.)

```
library(sotu)  # Load SOTU Dataset
sotu_info <- sotu::sotu_meta %>%
    filter(president %in% c("Dwight D. Eisenhower",
        "George Bush"))  # Get Info for Eisenhower and H.W.
head(sotu_info)  # Print Head
```

```
    X            president year years_active      party sotu_type
1 165 Dwight D. Eisenhower 1953    1953-1957 Republican   written
2 167 Dwight D. Eisenhower 1954    1953-1957 Republican    speech
3 168 Dwight D. Eisenhower 1955    1953-1957 Republican    speech
4 169 Dwight D. Eisenhower 1956    1953-1957 Republican    speech
5 170 Dwight D. Eisenhower 1956    1953-1957 Republican   written
6 171 Dwight D. Eisenhower 1957    1957-1961 Republican    speech
```

SOTU Address – Military (Cont.)

```r
indices <- c(sotu_info$X)  # Indices to Partition sotu_text

sotu_eisenhower_bush <- setNames(lapply(seq_len(nrow(sotu_info)),
    function(i) {
        cbind(sotu_info[i, ], text = sotu::sotu_text[[indices[i]]])
    }), paste0(sotu_info$president, " (", sotu_info$year,
    ")"))  # Nest Each Speech in List

names(sotu_eisenhower_bush)  # Print Names
```

```
 [1] "Dwight D. Eisenhower (1953)" "Dwight D. Eisenhower (1954)" "Dwight D. Eisenhower (1955)"
 [4] "Dwight D. Eisenhower (1956)" "Dwight D. Eisenhower (1956)" "Dwight D. Eisenhower (1957)"
 [7] "Dwight D. Eisenhower (1958)" "Dwight D. Eisenhower (1959)" "Dwight D. Eisenhower (1960)"
[10] "Dwight D. Eisenhower (1961)" "George Bush (1989)"          "George Bush (1990)"
[13] "George Bush (1991)"          "George Bush (1992)"
```

SOTU Address – Military (Cont.)

```r
military_words_regex <- paste0("(", paste(c("military",
    "army", "navy", "marines", "air force"), collapse = "|"),
    ")")  # 'Military' Words Regex
```

```
Dwight D. Eisenhower (1953)  --  12  Sentences
Dwight D. Eisenhower (1954)  --  15  Sentences
Dwight D. Eisenhower (1955)  --  17  Sentences
Dwight D. Eisenhower (1956)  --  0  Sentences
Dwight D. Eisenhower (1956)  --  9  Sentences
Dwight D. Eisenhower (1957)  --  8  Sentences
Dwight D. Eisenhower (1958)  --  25  Sentences
Dwight D. Eisenhower (1959)  --  11  Sentences
Dwight D. Eisenhower (1960)  --  6  Sentences
Dwight D. Eisenhower (1961)  --  7  Sentences
George Bush (1989)  --  4  Sentences
George Bush (1990)  --  6  Sentences
George Bush (1991)  --  1  Sentences
George Bush (1992)  --  2  Sentences
```

SOTU Address – Military (Cont.)

- Let's validate

```
unlist(sotu_eisenhower_bush[[14]]$military_text)  # Bush 1992 -- Print Example
```

```
     "Two years ago, I began planning cuts in military spending that reflected the changes of the new era
"The Secretary of Defense recommended these cuts after consultation with the Joint Chiefs of Staff. And I
```

SOTU Address – Military

- Sample appears to confirm that we're indeed recovering parts of the address related to the military.
- FWIW, certain policy elements are *always* in SOTU Addresses – e.g., the economy, education, and the military
- **What are some additional items we can use to capture rhetoric related to the `military`?**

SOTU Address – Military (Cont.)

```r
military_speeches <- data.frame()

for (i in 1:length(sotu_eisenhower_bush)) {
    temp_military <- unlist(sotu_eisenhower_bush[[i]]$military_text)
    if (length(temp_military) == 0) {
        next
    }
    temp_speech <- names(sotu_eisenhower_bush[i])
    temp_df <- data.frame(speech = temp_speech, military_text = temp_military)
    military_speeches <- bind_rows(military_speeches,
        temp_df)
}  # Combine to Single DF


military_speeches$president <- ifelse(grepl("Eisenhower",
    military_speeches$speech), "Eisenhower", "Bush")  # Add President ID

rownames(military_speeches) <- NULL
```

SOTU Address – Military (Cont.)

```
tibble(military_speeches)
```

```
# A tibble: 123 x 3
   speech                      military_text
   <chr>                       <chr>
 1 Dwight D. Eisenhower (1953) "But the problem of security demands closer cooperation among the nations
 2 Dwight D. Eisenhower (1953) "Europe's enlightened leaders have long been aware of these facts. All the
 3 Dwight D. Eisenhower (1953) "The needed unity of Western Europe manifestly cannot be manufactured from
 4 Dwight D. Eisenhower (1953) "This war is, for Americans, the most painful phase of Communist aggressio
 5 Dwight D. Eisenhower (1953) "This has meant, in effect, that the United States Navy was required to se
 6 Dwight D. Eisenhower (1953) "Consequently there is no longer any logic or sense in a condition that re
 7 Dwight D. Eisenhower (1953) "Our problem is to achieve adequate military strength within the limits of
 8 Dwight D. Eisenhower (1953) "Both military and economic objectives demand a single national military p
 9 Dwight D. Eisenhower (1953) "We must not let traditions or habits of the past stand in the way of deve
10 Dwight D. Eisenhower (1953) "Because of the complex technical nature of our military organization and
# i 113 more rows
```

SOTU Address – Exercise

Using the `sotu` dataset, let's analyze another pairing of executives and policy area.

- Select another pair of presidents (since 1960)
- Select another policy area (ex: the economy, civil rights, energy, etc.)
- Develop a regular vocabulary and regular expression (regex) to capture parts of the speeches discussing those policy areas.
- Validate your data collection by sampling a few elements of the collecte data

High-Dimensional Text

- **Recall**: Using text data often relieves concerns re: small observations **but** high-dimensionality often introduces sparsity problems of its own.
- As the number of unique **tokens** (word units) increase, observations become sparse and distances between documents become less informative.

Normalizing Text

- **Complexity Reduction**: Process of systematically transforming raw text to reduce the number and variability of unique tokens (features) while preserving meaningful semantic content.
- Reducing feature complexity constrains the hypothesis space, improving generalization to new documents.
- **Normalization** (lowercasing, lemmatization, stopword removal, etc.) reduces dimensionality and stabilizes similarity measures.

Normalizing Text – Big Consideration

- Denny & Spirling (2018) – Main point?

Normalizing Text – Big Consideration

- Denny & Spirling (2018) – Main point?
- Pre-processing – *if & how* – can have fundamental impact on results.
- Week 4 Problem Set: How you partitioned text impacted your summary values
- Models produce values/estimates given observational data – how (and how much) your input is structured will have an impact on output!

Complexity Reduction – R Function

```r
reduce_complexity <- function(text) {
    text <- tolower(text)  # Lower Case
    text <- tm::removePunctuation(text)  # Punctuation
    text <- tm::removeNumbers(text)  # Numbers
    text <- tm::removeWords(text, tm::stopwords("english"))  # Stop Words
    text <- unlist(stringr::str_split(text, "\\s+"))  # Tokenize
    text <- textstem::lemmatize_words(text)  # Lemmatize
    text <- paste(text, collapse = " ")  # Re-Append
    text <- gsub("\\s{2,}", " ", text)  # 2 or More Spaces --> One Space
    text <- trimws(text)  # White Space
    return(text)
}  # Function to Process Text for Bag of Words
```

Complexity Reduction – Comparison

```r
regular <- military_speeches$military_text[1]  # Print Regular Text
normalized <- reduce_complexity(military_speeches$military_text[1])  # Processed Text Example

cat(regular)
```

```
But the problem of security demands closer cooperation among the nations of Europe than has been known to date. Only
a more closely integrated economic and political system can provide the greatly increased economic strength needed to
maintain both necessary military readiness and respectable living standards.
```

```r
cat(normalized)
```

```
problem security demand close cooperation among nation europe know date closely integrate economic political system
can provide greatly increase economic strength need maintain necessary military readiness respectable live standard
```

Complexity Reduction – Exercise

- Using `reduce_complexity()` function, recover text from your example SOTU policy.
- *Note*: Goals of this pre-processing step are to preserve information while reducing noise – do you think that's still the case once you've normalized your text?

Document Frequency Matrix

- **DFM**: A *sparse* matrix where rows represent documents and columns represent features (usually word types), and each cell contains the frequency of that feature in that document.
- **Recall**: Our first step to analyze text at the document unit was to create a corpus of text – we will do the same then convert that corpus into a sparse matrix using `quanteda`.

Creating a DFM – Create a Corpus First!

```r
military_speeches <- military_speeches %>%
    mutate(military_text_clean = sapply(military_text,
        reduce_complexity))  # Apply Complexity Reduction

sotu_corpus <- quanteda::corpus(military_speeches,
    text_field = "military_text_clean")  # Convert to Corpus Object

sotu_tokens <- quanteda::tokens(sotu_corpus)  # Recover Tokens from Corpus Object

sotu_dfm <- dfm(sotu_tokens) %>%
    dfm_trim(min_termfreq = 2)  # Convert to DFM -- Remove Words w/ Less Than 2 Appearances
```

Creating a DFM (Cont.)

```r
quanteda::topfeatures(sotu_dfm, 20)  # 20-top Features (Words)
```

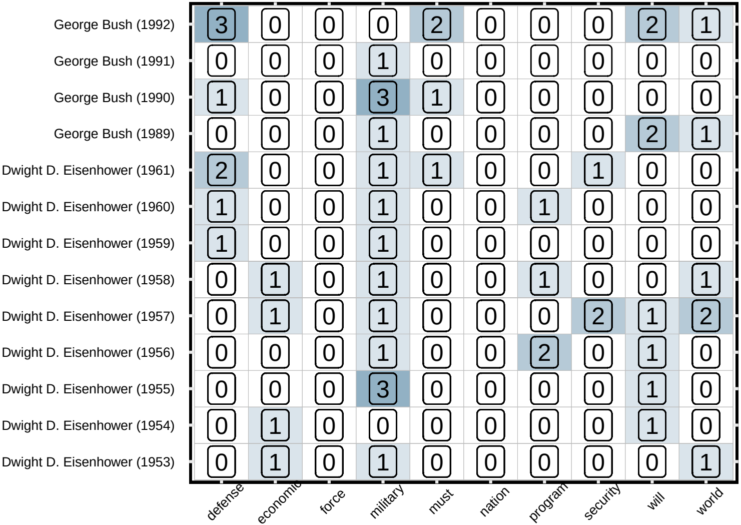
```
military     will  defense     must    force   nation security economic  program    world strength       ma
     132       65       50       44       43       40       38       34       34       32       31
    year      new    power    peace     need maintain    shall
      28       26       24       24       23       22       22
```

Visualizing the DFM – Heatmap

```r
sotu_dfm_reduced <- sotu_dfm[, names(topfeatures(sotu_dfm, 10))] # Filter to Top-20 Terms
speech_labels <- docvars(sotu_dfm_reduced, "speech")

sotu_dfm_reduced %>%
  quanteda::convert(to = "data.frame") %>% # Convert DFM to DF
  mutate(speech = speech_labels) %>% # Append Speech Labels
  tidyr::pivot_longer(cols = -c(doc_id, speech), names_to = "term", values_to = "frequency") %>%
  ggplot(aes(x = term, y = speech, fill = frequency)) +
  geom_tile(colour = 'grey') +
  geom_label(aes(label = frequency)) +
  scale_fill_gradient(low = "white", high = "deepskyblue4") +
  theme_minimal() +
  labs(x = "\nTerm", y = "Speech\n") +
  default_ggplot_theme
```

Visualizing the DFM – Heatmap



	defense	economic	force	military	must	nation	program	security	will	world
George Bush (1992)	3	0	0	0	2	0	0	0	2	1
George Bush (1991)	0	0	0	1	0	0	0	0	0	0
George Bush (1990)	1	0	0	3	1	0	0	0	0	0
George Bush (1989)	0	0	0	1	0	0	0	0	2	1
Dwight D. Eisenhower (1961)	2	0	0	1	1	0	0	1	0	0
Dwight D. Eisenhower (1960)	1	0	0	1	0	0	1	0	0	0
Dwight D. Eisenhower (1959)	1	0	0	1	0	0	0	0	0	0
Dwight D. Eisenhower (1958)	0	1	0	1	0	0	1	0	0	1
Dwight D. Eisenhower (1957)	0	1	0	1	0	0	0	2	1	2
Dwight D. Eisenhower (1956)	0	0	0	1	0	0	2	0	1	0
Dwight D. Eisenhower (1955)	0	0	0	3	0	0	0	0	1	0
Dwight D. Eisenhower (1954)	0	1	0	0	0	0	0	0	1	0
Dwight D. Eisenhower (1953)	0	1	0	1	0	0	0	0	0	1

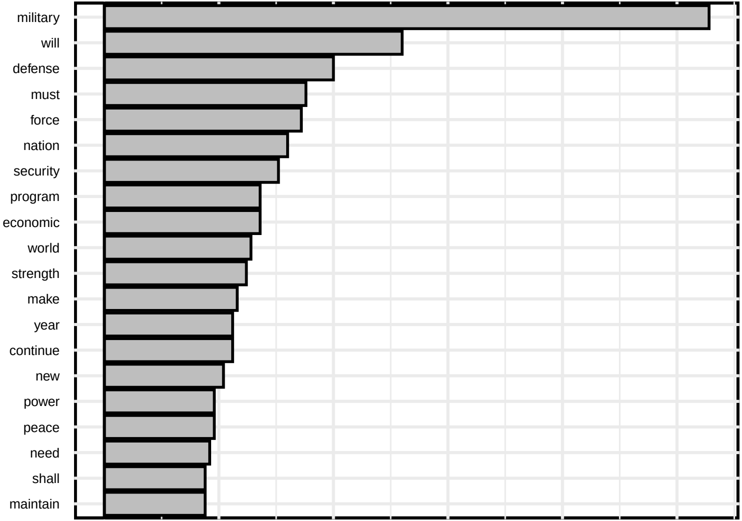
Visualizing the DFM – Top Terms Bar Plot

```
top_terms <- topfeatures(sotu_dfm, 20)

sotu_bar_df <- data.frame(term = names(top_terms),
    frequency = as.numeric(top_terms))

sotu_bar_df %>%
    ggplot(aes(x = frequency, y = reorder(term, frequency))) +
    geom_col(fill = "grey", colour = "black") + labs(x = "\nFrequency",
    y = "Term\n") + geom_vline(xintercept = 0) + scale_x_continuous(breaks = seq(25,
    150, 25)) + default_ggplot_theme
```

Visualizing the DFM – Top Terms Bar Plot

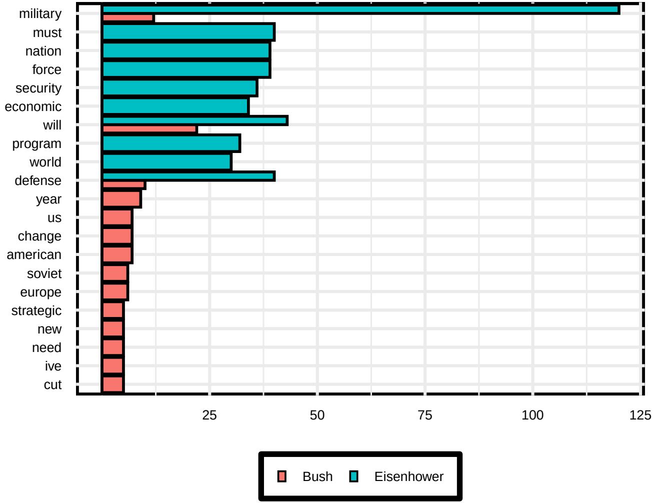


Visualizing the DFM – Top Terms Bar Plot (By Pres)

```r
sotu_term_freq <- textstat_frequency(sotu_dfm, group = president)

sotu_term_freq %>%
  group_by(group) %>%
  slice_max(frequency, n = 10) %>% # Take top-10 Terms
  ggplot(aes(y = reorder(feature, frequency), x = frequency)) +
  geom_col(aes(fill = group), colour = 'black', position = position_dodge()) +
  scale_x_continuous(breaks = seq(25, 125, 25)) +
  geom_vline(xintercept = 0) +
  default_ggplot_theme
```

Visualizing the DFM – Top Terms Bar Plot (By Pres)



Visualization Exercise

Your turn – Use your custom policy area to replicate the three visualizations.

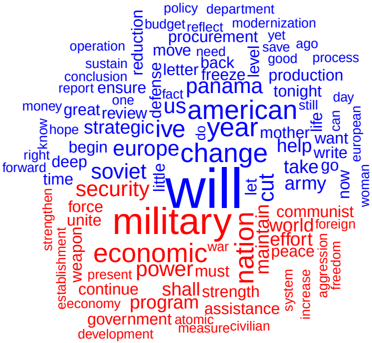
WordClouds

```
president_dfm <- dfm_group(sotu_dfm, groups = military_speeches$president)  # Group DFM by President

quanteda.textplots::textplot_wordcloud(president_dfm,
    comparison = TRUE, max_words = 100, color = c("blue",
        "red"))
```

WordClouds

Bush



Eisenhower

Next Class

- Modeling the Bag of Words – Dictionaries, Multinomial Language Model, and Vector Space Model
- **Reminder**: Class 6 Problem Set Due Sunday
- **Reminder**: Respond to Final Project Notes!