Lecture 9: Text As Data and Dictionaries

Jack Blumenau

THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Today’s lecture

- Introduction to Quantitative Text Analysis
- Representing Text as Data
- Dictionaries
- Application
- Validation
- Conclusion

Introduction to Quantitative Text Analysis

Much of the Social World is Textual

Language is central to almost all social interaction

1. Laws are written ✒️
2. Political events are discussed 📢
3. History is recorded 📚
4. People communicate ✉️

But these interactions have not been amenable to quantitative analysis until recently.

THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

The Growth of Quantitative Text Analysis

Two major changes that contributed to the growth of QTA:

1. Enormous increase in *availability of digitized texts*
2. Development of *powerful and easily applicable methods*

Consequence: we have the ability now to interrogate central questions in social science using data that was never available in the past.

Quantitative Text Analysis

We will be thinking about different methods of doing one core thing:

Assigning numbers to words and documents in order to measure latent concepts in text.

Although the methods we use to generate these numbers differ, the common goal will be to assign numbers that enable us measure latent concepts from large corpora of text.

Assigning Numbers to Words/Documents

Applications in QTA

- How does the media cover the economy?
- When did Western political culture diverge from the rest of the world?
- How do central bankers make decisions on economic policy?
- How has the cultural meaning of words changed over time?
- How can we detect online hate speech?
- Which interest groups have policy influence?

Measuring Interest Group Influence Using Quantitative Text Analysis

Heike Klüver
University of Mannheim, Germany

ABSTRACT

The analysis of interest group influence is crucial in order to explain policy outcomes and to assess the democratic legitimacy of the European Union. However, owing to methodological difficulties in operationalizing influence, only few have studied it. This article therefore proposes a new approach to the measurement of influence, drawing on quantitative text analysis. By comparing interest groups' policy positions with the final policy output, one can draw conclusions about the winners and losers of the decision-making process. In order to examine the appl f text

Assumptions

Many of these approaches share a set of common **assumptions**:

1. *Texts represent* observable manifestations of *underlying characteristics* of interest (usually attribute of authors)
2. Texts *can be represented* through extracting their *features* (for now, words)
3. *Analysis* of those features can *produce meaningful estimates* of the underlying characteristic of interest

For any given application, these assumptions may or may not be met.

Principles of Quantitative Text Analysis

Workflow

Each quantitative text analysis follows a similar workflow:

1. **Conversion** of textual features into a quantitative matrix
2. A **quantitative or statistical procedure** to extract information from the quantitative matrix
3. **Summary** and interpretation of the quantitative results

Workflow

An economic miracle is taking place in the United States, and the only thing that can stop it are foolish wars, politics, or ridiculous partisan investigations.

The United States of America right now has the strongest, most durable economy in the world. We're in the middle of the longest streak of private sector job creation in history.

We reinvented Government, transforming it into a catalyst for new ideas that stress both opportunity and responsibility and give our people the tools they need to solve their own problems.

To build a prosperd future, we must trus people with their own money and empower them to grow our economy.

Source texts

Processed text as a document-feature matrix

documents	economy	united	wall	crime	climate
	features				
Clinton-2000	10	4	1	5	1
Bush-2008	6	4	0	0	1
Obama-2016	16	4	1	0	4
Trump-2019	5	19	6	2	0

Quantitative analysis and inference

Describing texts quantitatively or stylistically
Identifying keywords
Measuring ideology or sentiment in documents
Mapping semantic networks
Identifying topics and estimating their prevalence
Measuring document or term similarities
Classifying documents

Workflow

In reality, there are additional steps:

1. Select Documents
2. Digitize documents
3. Represent as quantitative data
4. Analyse data
5. Validate analysis
6. Interpret analysis

Representing Text as Data

Motivating Example

Motivating Example

The UN Sustainable Development Goals are a set of 17 connected global goals which represent “a shared blueprint for peace and prosperity” for people across the world. Each goal is associated with a series of specific targets and indicators.

Question: (How) can we characterise the UN Sustainable Development Goals as numeric data?

```r
sdg <- read.csv("data/SDG-goals.csv")
sdg$description
```

```
 [1] "End poverty in all its forms everywhere"
 [2] "End hunger, achieve food security and improved nutrition and promote sustainable agriculture"
 [3] "Ensure healthy lives and promote well-being for all at all ages"
 [4] "Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all"
 [5] "Achieve gender equality and empower all women and girls"
 [6] "Ensure availability and sustainable management of water and sanitation for all"
 [7] "Ensure access to affordable, reliable, sustainable and modern energy for all"
 [8] "Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent
work for all"
 [9] "Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation"
[10] "Reduce inequality within and among countries"
[11] "Make cities and human settlements inclusive, safe, resilient and sustainable"
[12] "Ensure sustainable consumption and production patterns"
[13] "Take urgent action to combat climate change and its impacts"
[14] "Conserve and sustainably use the oceans, seas and marine resources for sustainable development"
[15] "Protect, restore and promote sustainable use of terrestrial ecosystems, sustainably manage forests, combat
desertification, and halt and reverse land degradation and halt biodiversity loss"
[16] "Promote peaceful and inclusive societies for sustainable development, provide access to ju[illegible]r all
and build effective, accountable and inclusive institutions at all levels"
[17] "Strengthen the means of implementation and revitalize the global partnership for sustainab[illegible]opment"
```

Motivating Example

```r
sdg$long_description
```

```
 [1] "End poverty in all its forms everywhere By 2030, eradicate extreme poverty for all people everywhere, currently measured as people living on less than $1.25 a day By 2030, reduce at least by half the proportion of men, women and children of all ages living in poverty in all its dimensions according to national definitions Implement nationally appropriate social protection systems and measures for all, including floors, and by 2030 achieve substantial coverage of the poor and the vulnerable  By 2030, ensure that all men and women, in particular the poor and the vulnerable, have equal rights to economic resources, as well as access to basic services, ownership and control over land and other forms of property, inheritance, natural resources, appropriate new technology and financial services, including microfinance By 2030, build the resilience of the poor and those in vulnerable situations and reduce their exposure and vulnerability to climate-related extreme events and other economic, social and environmental shocks and disasters  Ensure significant mobilization of resources from a variety of sources, including through enhanced development cooperation, in order to provide adequate and predictable means for developing countries, in particular least developed countries, to implement programmes and policies to end poverty in all its dimensions  Create sound policy frameworks at the national, regional and international levels, based on pro-poor and gender-sensitive development strategies, to support accelerated investment in poverty eradication actions "
 [2] "End hunger, achieve food security and improved nutrition and promote sustainable agriculture By 2030, end hunger and ensure access by all people, in particular the poor and people in vulnerable situations, including infants, to safe, nutritious and sufficient food all year round By 2030, end all forms of malnutrition, including achieving, by 2025, the internationally agreed targets on stunting and wasting in children under 5 years of age, and address the nutritional needs of adolescent girls, pregnant and lactating women and older persons By 2030, double the agricultural productivity and incomes of small-scale food producers, in particular women, indigenous peoples, family farmers, pastoralists and fishers, including through secure and equal access
```

There Is No Single Right Way To Represent Text

Which features of text would be most helpful for the following research questions?

1. Predicting whether the author of a text message was young or old
2. Measuring the financial content of news coverage
3. Assessing the complexity of a piece of writing

There Is No Single Right Way To Represent Text

Which features of text would be most helpful for the following research questions?

1. Predicting whether the author of a text message was young or old
    - Emojis; informal language; length
2. Measuring the financial content of news coverage
    - Words relating to finance
3. Assessing the complexity of a piece of writing
    - Number of sylables; relative number of adjectives, nouns, verbs, etc

Implication: feature selection will depend on your research question.

Document-feature matrix

Document-Feature Matrix (DFM)

A document-feature matrix is a common way of representing text data in quantitative form.

- The **rows** of the matrix indicate the **documents**.
- The **columns** of the matrix indicate the **features** (words, etc).

DFM’s are *parsimonious* representations which discard information. But they are helpful!

In order to construct a dfm, we need to made decisions about both documents and features.

Terminology

Document

Basic unit (text) of analysis

Corpus

A structured set of documents for analysis

Type

A unique feature in the corpus e.g. a word (“flies”), a punctuation mark, a part-of-speech

Token

An instance of a type in a document e.g. the occurrence of the word in a given document

Selecting documents

Selecting documents is an important, and often ignored, step in any QTA analysis.

Key questions:

1. Is it possible/feasible to collect a set of documents?
2. Is the corpus representative of the population of interest?
3. Is it ethical to examine documents of this sort at scale?

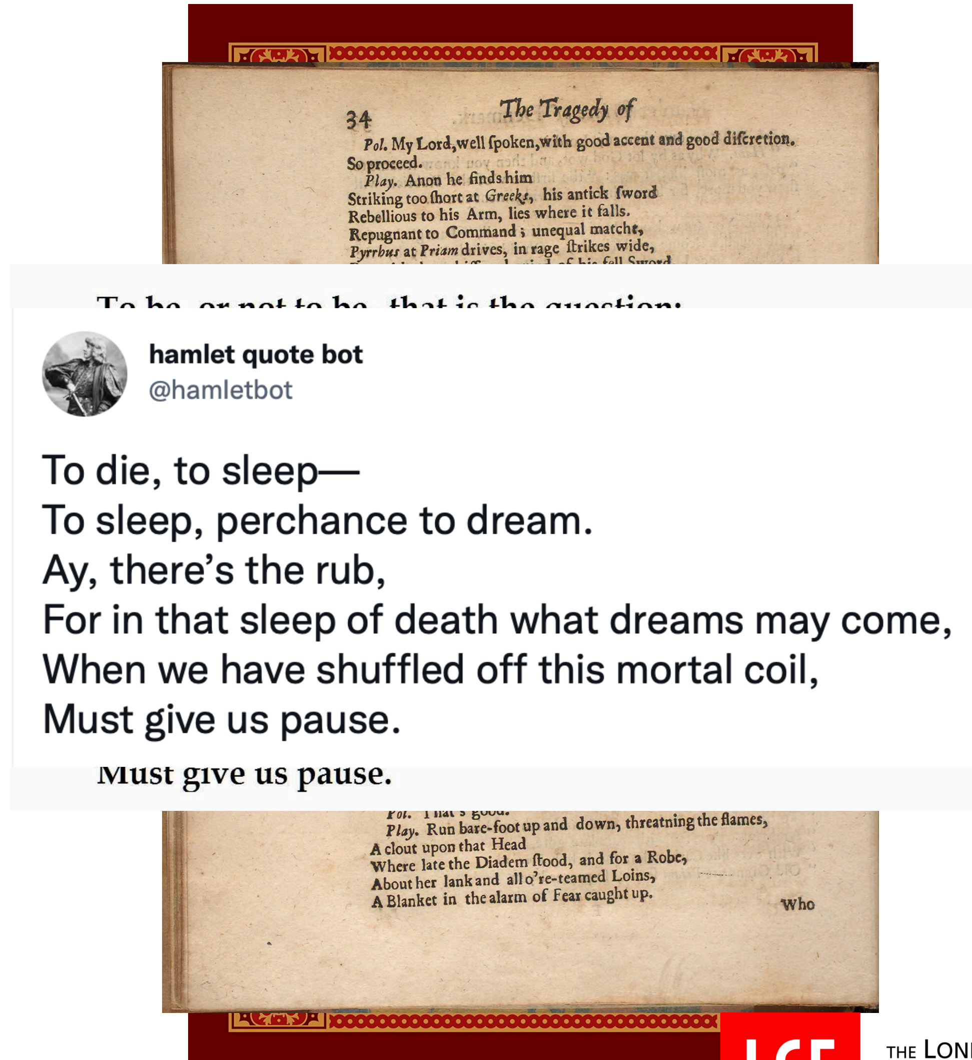
Implication: The selection of texts is consequential to the conclusions we can draw.

Strategies for defining "documents"

A "document" is the typical unit of analysis in QTA. But what is a document?

- Entire document
- Pages
- Paragraphs
- Tweets

Key: Depends on the research question.



Strategies for defining “features”

- Words
- N-grams
- Language sequences
  - Parts of speech
  - Named entities
  - Dependency parsing
- Word segments, especially for languages using compound words, e.g.

  Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz

Strategies for defining "features"

- Words
- N-grams
- Language sequences
  - Parts of speech
  - Named entities
  - Dependency parsing
- Word segments, especially for languages using compound words, e.g.

  Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz

  The law concerning the delegation of duties for the supervision of cattle marking and the labelling of beef

Bags of words

1. The simplest possible way of characterising a corpus is by counting words
2. For each text, we record how many times each unique word appears
3. We ignore everything else.

Bags of words assumptions

1. The words in a document convey meaning
2. Word order does not matter
3. Word combinations do not matter (i.e. negation)
4. Grammar does not matter
5. Words are the *only* relevant features (not punctuation, not syllables, etc)

The importance of these assumptions depends on the application.

Bag of words assumption

1. Time flies like an arrow.
2. Fruit flies like a banana.

	time	flies	fruit	like	an	a	banana	arrow
Sentence 1	1	1	0	1	1	0	0	1
Sentence 2	0	1	1	1	0	1	1	0

- The dependency structure between words in each sentence is lost
- The word “flies” has two different meanings (metaphorical versus literal)
- The word “like” has two different meanings (preposition versus verb)
- The “joke” is no longer funny

Bags of words

```r
# Load the quanteda library
library(quanteda)

# Convert the sdg data.frame into a corpus
sdg_corpus <- corpus(sdg, text_field = "long_description")

# Take the corpus
sdg_dfm <- sdg_corpus %>%
           # Tokenize (split) the corpus into individual words
           tokens() %>%
           # Construct a document-feature matrix
           dfm()

# Print the dfm
sdg_dfm
```

```
Document-feature matrix of: 17 documents, 1,085 features (86.41% sparse) and 2 docvars.
       features
docs    end poverty in all its forms everywhere by 2030  ,
  text1   2       5  9   7   3     2          2  6    5 24
  text2   3       0 11   5   0     2          0  7    4 43
  text3   2       0  7   7   0     0          0  8    6 33
  text4   0       0  6   8   0     0          0  9    8 39
  text5   1       0  5   9   0     3          1  0    0 11
  text6   1       0  3   5   0     0          0  8    6 21
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 1,075 more features ]
```

Coding Interlude

Wait, what is this %>% thing?

- This is called a “pipe”
- It takes the output of one function and passes it to another function

E.g.

```r
my_vector <- c(1,2,3)
mean(my_vector) %>% sqrt()
```

```
[1] 1.414214
```

Bags of words

How many features are there in this dfm?

```r
ncol(sdg_dfm)
```

```
[1] 1085
```

And how many documents?

```r
nrow(sdg_dfm)
```

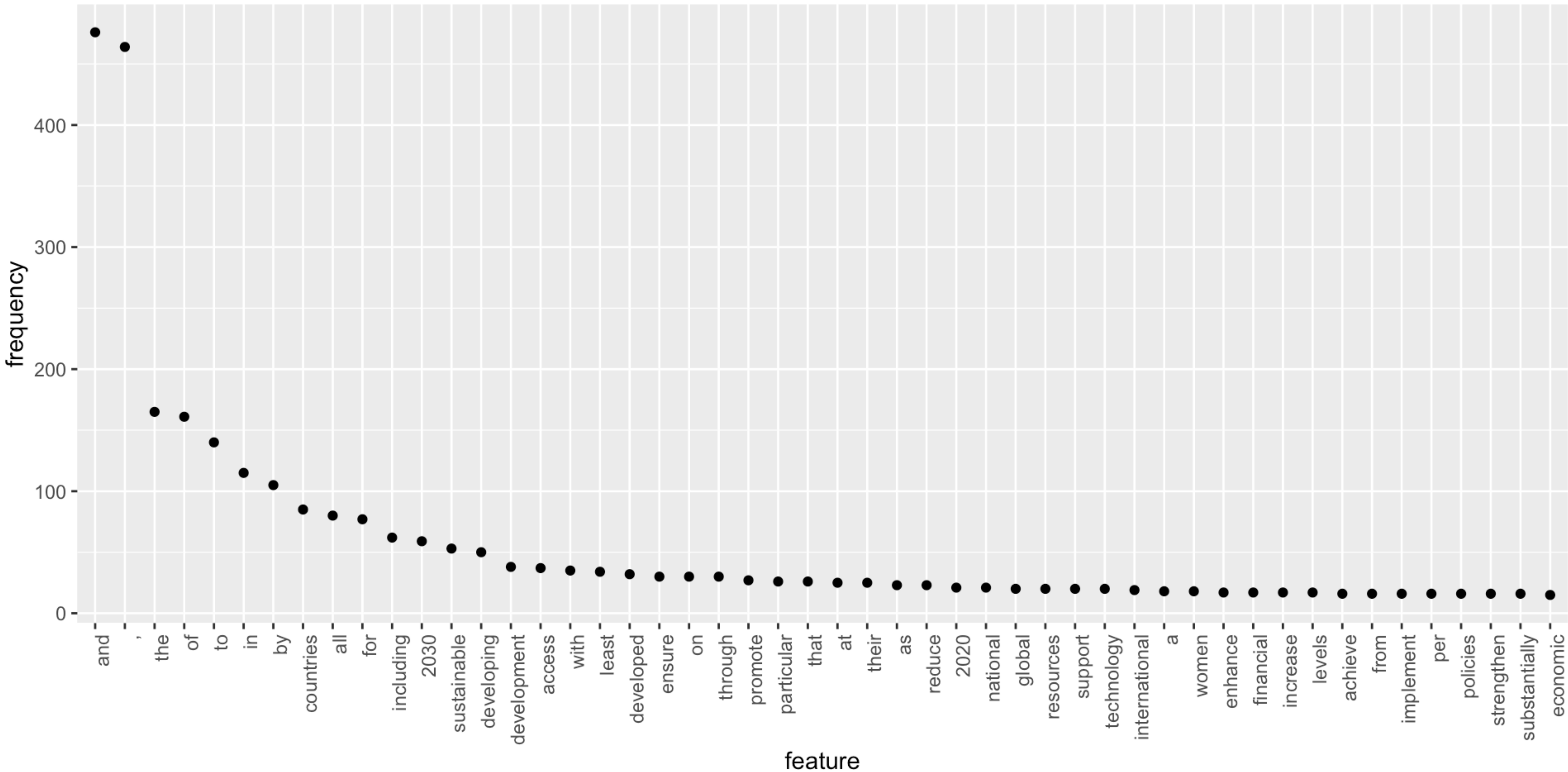
```
[1] 17
```

And what are the most common features in this dfm?

```r
topfeatures(sdg_dfm, 10)
```

```
      and         ,       the        of        to        in        by countries
      476       464       165       161       140       115       105        85
      all       for
       80        77
```

Top features

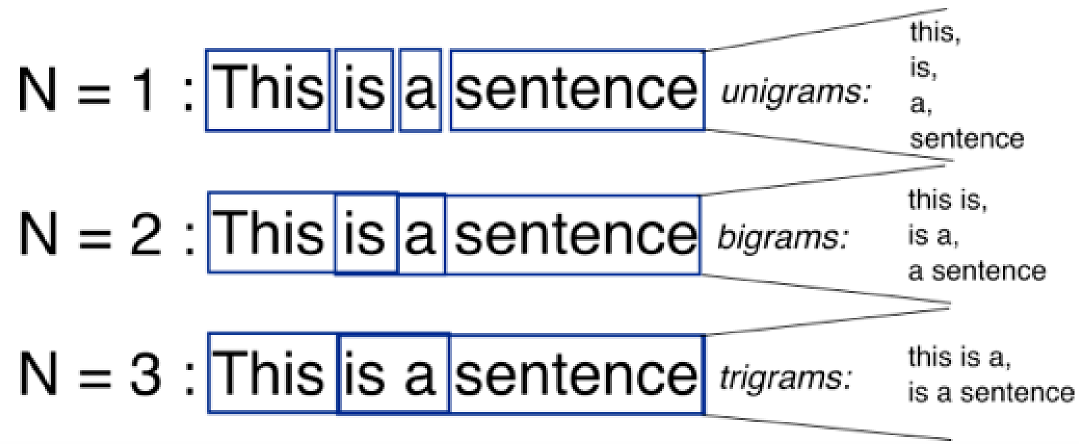


THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Word sequences/N-grams

N-grams

Contiguous sequence of words from document (1-gram, unigram; 2-gram, bigram)



Word sequences/N-grams

```
sdg_dfm <- sdg_corpus %>%
                    # Split the corpus into individual words
                    tokens() %>%
                    # Construct a document-feature matrix
                    dfm()

sdg_dfm
```

Word sequences/N-grams

```r
sdg_dfm_bigram <- sdg_corpus %>%
                    # Split the corpus into individual words
                    tokens() %>%
                    # Construct uni-grams and bi-grams
                    tokens_ngrams(1:2) %>%
                    # Construct a document-feature matrix
                    dfm()

sdg_dfm_bigram
```

```
Document-feature matrix of: 17 documents, 4,337 features (90.46% sparse) and 2 docvars.
       features
docs    end poverty in all its forms everywhere by 2030  ,
  text1   2       5  9   7   3     2          2  6    5 24
  text2   3       0 11   5   0     2          0  7    4 43
  text3   2       0  7   7   0     0          0  8    6 33
  text4   0       0  6   8   0     0          0  9    8 39
  text5   1       0  5   9   0     3          1  0    0 11
  text6   1       0  3   5   0     0          0  8    6 21
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 4,327 more features ]
```

Word sequences/N-grams

```r
sdg_dfm_trigram <- sdg_corpus %>%
                    # Split the corpus into individual words
                    tokens() %>%
                    # Construct uni-grams, bi-grams and tri-grams
                    tokens_ngrams(1:3) %>%
                    # Construct a document-feature matrix
                    dfm()

sdg_dfm_trigram
```

```
Document-feature matrix of: 17 documents, 8,685 features (91.86% sparse) and 2 docvars.
       features
docs    end poverty in all its forms everywhere by 2030  ,
  text1   2       5  9   7   3     2          2  6    5 24
  text2   3       0 11   5   0     2          0  7    4 43
  text3   2       0  7   7   0     0          0  8    6 33
  text4   0       0  6   8   0     0          0  9    8 39
  text5   1       0  5   9   0     3          1  0    0 11
  text6   1       0  3   5   0     0          0  8    6 21
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 8,675 more features ]
```

Word sequences/N-grams

How many features are there in these dfms?

```
ncol(sdg_dfm)
```

```
[1] 1085
```

```
ncol(sdg_dfm_bigram)
```

```
[1] 4337
```

```
ncol(sdg_dfm_trigram)
```

```
[1] 8685
```

Strategies for feature selection

- This can lead to a lot of features!
- For this example (very small) corpus:
  - 17 documents
  - 1085 unique words
  - 4337 unique 1-gram and 2-gram sequences
  - 8685 unique 1-gram, 2-gram and 3-gram sequences
- The resulting dfms are also very *sparse* – they contain a high fraction of zeros because most n-grams do not appear in most documents

```r
sparsity(sdg_dfm)
```

```
[1] 0.8640824
```

```r
sparsity(sdg_dfm_bigram)
```

```
[1] 0.9046237
```

```r
sparsity(sdg_dfm_trigram)
```

```
[1] 0.9186359
```

Strategies for feature selection

1. Reduce complexity
   - Convert to lowercase (automatic in `quanteda`), remove punctuation (not automatic in `quanteda`)
2. Deliberate disregard
   - Ignore words that have no substantive content ("stop" words)
3. Word stemming/lematization
   - Define some words as equivalent to each other (school, schools, schooling, etc)
4. Filter by frequency
   - Document frequency: Ignore words that occur rarely across documents
   - Term frequency: Ignore words that occur rarely overall
5. Purposive selection
   - Select only certain words to analyse

Common stop words

```
1  stopwords("en")
```

```
  [1] "i"          "me"         "my"         "myself"     "we"
  [6] "our"        "ours"       "ourselves"  "you"        "your"
 [11] "yours"      "yourself"   "yourselves" "he"         "him"
 [16] "his"        "himself"    "she"        "her"        "hers"
 [21] "herself"    "it"         "its"        "itself"     "they"
 [26] "them"       "their"      "theirs"     "themselves" "what"
 [31] "which"      "who"        "whom"       "this"       "that"
 [36] "these"      "those"      "am"         "is"         "are"
 [41] "was"        "were"       "be"         "been"       "being"
 [46] "have"       "has"        "had"        "having"     "do"
 [51] "does"       "did"        "doing"      "would"      "should"
 [56] "could"      "ought"      "i'm"        "you're"     "he's"
 [61] "she's"      "it's"       "we're"      "they're"    "i've"
 [66] "you've"     "we've"      "they've"    "i'd"        "you'd"
 [71] "he'd"       "she'd"      "we'd"       "they'd"     "i'll"
 [76] "you'll"     "he'll"      "she'll"     "we'll"      "they'll"
 [81] "isn't"      "aren't"     "wasn't"     "weren't"    "hasn't"
 [86] "haven't"    "hadn't"     "doesn't"    "don't"      "didn't"
 [91] "won't"      "wouldn't"   "shan't"     "shouldn't"  "can't"
 [96] "cannot"     "couldn't"   "mustn't"    "let's"      "that's"
[101] "who's"      "what's"     "here's"     "there's"    "when's"
[106] "where's"    "why's"      "how's"      "a"          "an"
```

But no list should be considered universal…

THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Other common stop words

```
1  stopwords("smart")
```

```
 [1] "a"             "a's"           "able"          "about"
 [5] "above"         "according"     "accordingly"   "across"
 [9] "actually"      "after"         "afterwards"    "again"
[13] "against"       "ain't"         "all"           "allow"
[17] "allows"        "almost"        "alone"         "along"
[21] "already"       "also"          "although"      "always"
[25] "am"            "among"         "amongst"       "an"
[29] "and"           "another"       "any"           "anybody"
[33] "anyhow"        "anyone"        "anything"      "anyway"
[37] "anyways"       "anywhere"      "apart"         "appear"
[41] "appreciate"    "appropriate"   "are"           "aren't"
[45] "around"        "as"            "aside"         "ask"
[49] "asking"        "associated"    "at"            "available"
[53] "away"          "awfully"       "b"             "be"
[57] "became"        "because"       "become"        "becomes"
[61] "becoming"      "been"          "before"        "beforehand"
[65] "behind"        "being"         "believe"       "below"
[69] "beside"        "besides"       "best"          "better"
[73] "between"       "beyond"        "both"          "brief"
[77] "but"           "by"            "c"             "c'mon"
[81] "c's"           "came"          "can"           "can't"
[85] "cannot"        "cant"          "cause"         "causes"
```

Stop words example

End poverty in all its forms everywhere

End hunger, achieve food security and improved nutrition and promote sustainable agriculture

Ensure healthy lives and promote well-being for all at all ages

Stop words example

End poverty ~~in all its~~ forms everywhere

End hunger, achieve food security ~~and~~ improved nutrition ~~and~~ promote sustainable agriculture

Ensure healthy lives ~~and~~ promote well-being ~~for all at all~~ ages

Stop words can matter

Compare…

> It was a nice party, Pablo had brought his ukulele.

To…

> It was a nice party, **but** Pablo had brought his ukulele.

	nice	party	Pablo	brought	ukulele
Sentence 1	1	1	1	1	1
Sentence 2	1	1	1	1	1

Removing stop words in R

```
sdg_dfm <- sdg_corpus %>%
           tokens() %>%
           dfm()

sdg_dfm
```

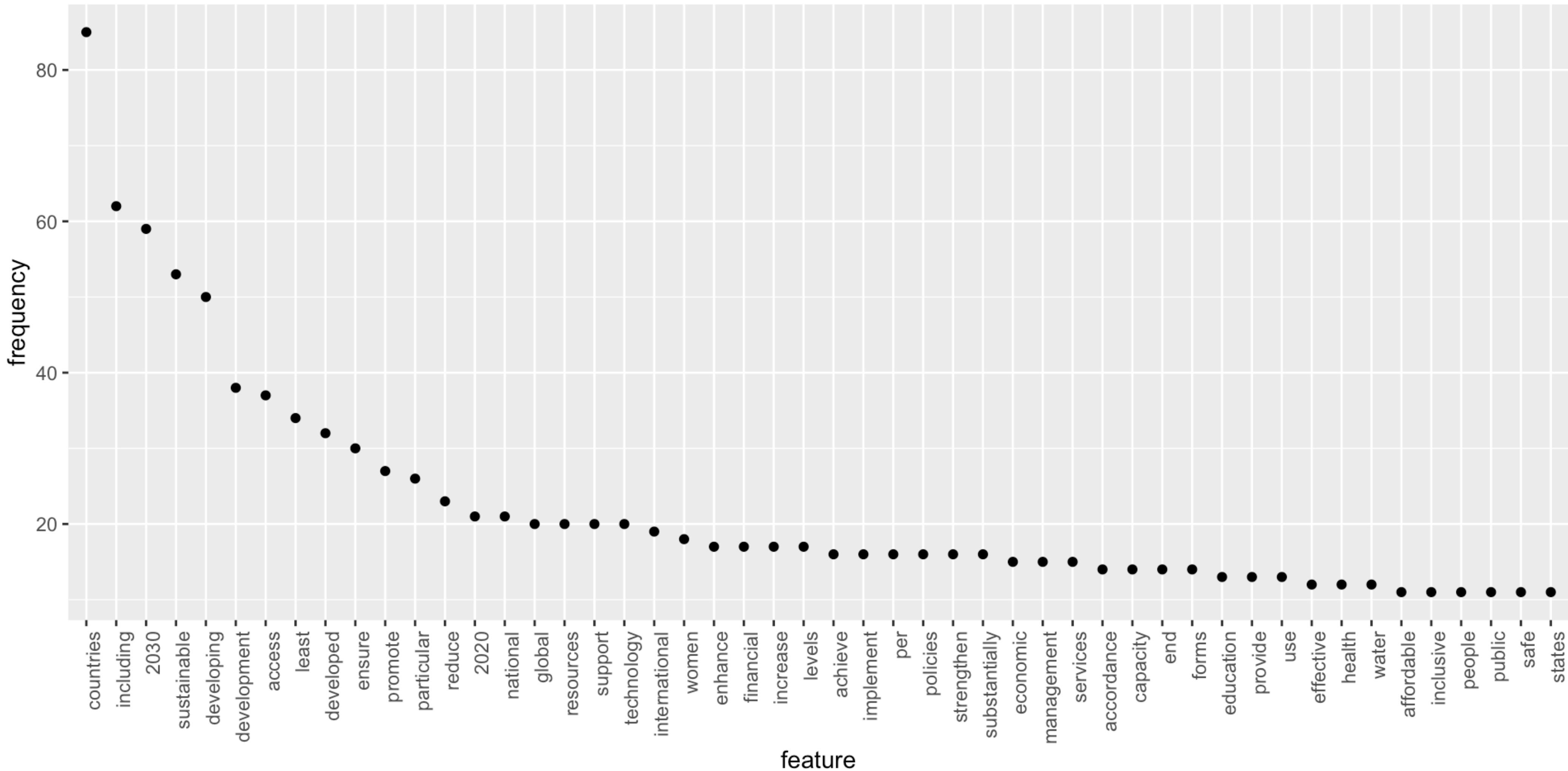
Removing stop words in R

```
sdg_dfm_no_stop <- sdg_corpus %>%
           tokens(remove_punct = TRUE) %>%
           tokens_remove(stopwords("en")) %>%
           dfm()

sdg_dfm_no_stop
```

```
Document-feature matrix of: 17 documents, 1,034 features (87.62% sparse) and 2 docvars.
       features
docs    end poverty forms everywhere 2030 eradicate extreme people currently
  text1   2       5     2          2    5         1       2      2         1
  text2   3       0     2          0    4         0       2      2         0
  text3   2       0     0          0    6         0       0      0         0
  text4   0       0     0          0    8         0       0      0         0
  text5   1       0     3          1    0         0       0      0         0
  text6   1       0     0          0    6         0       0      1         0
       features
docs    measured
  text1        1
  text2        0
  text3        0
  text4        0
  text5        0
  text6        0
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 1,024 more features ]
```

Top features



Stemming and lematization

Stemming

Process for reducing inflected (or sometimes derived) words to their stem, base or root form. Stemmers operate on single words without knowledge of the context.

Example:

Production, producer, produce, produces, produced → produc

Lemmatization

Algorithmic process of converting words to their lemma forms.

Example:

am, are, is → be

Stemming is a crude heuristic process that chops off the ends of words. Lemmatization is smarter, but slower.

Stemming example

End poverty in all its forms everywhere

End hunger, achieve food security and improved nutrition and promote sustainable agriculture

Ensure healthy lives and promote well-being for all at all ages

Stemming example

End poverti in all it form everywher

End hunger , achiev food secur and improv nutrit and promot sustain agricultur

Ensure healthi live and promot well-b for all at all age

Stemming in R

```
sdg_dfm <- sdg_corpus %>%
           tokens() %>%
           dfm()

sdg_dfm
```

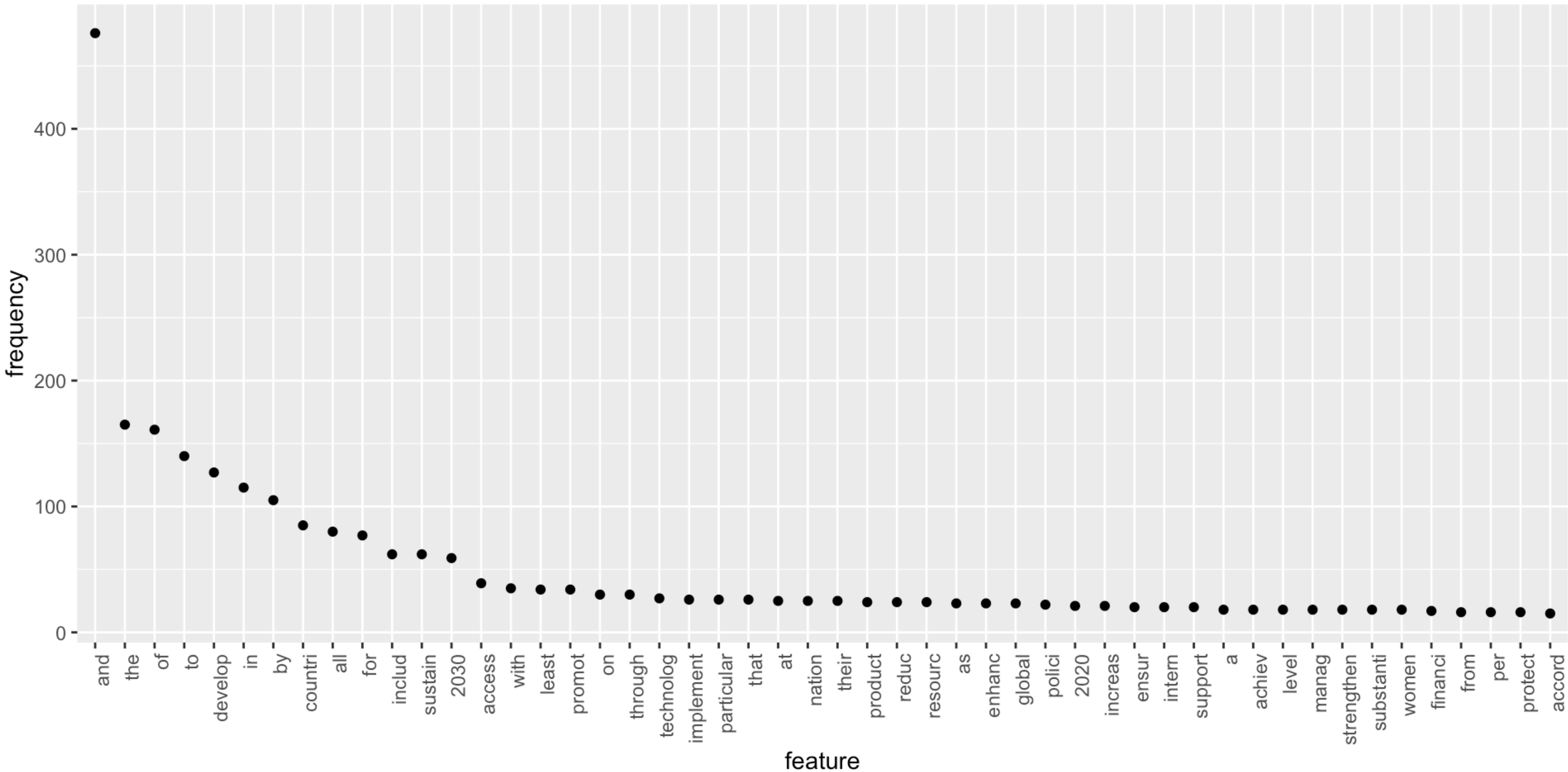
Stemming in R

```r
sdg_dfm_stem <- sdg_corpus %>%
            tokens(remove_punct = TRUE) %>%
            tokens_wordstem() %>%
            dfm()

sdg_dfm_stem
```

```
Document-feature matrix of: 17 documents, 872 features (84.37% sparse) and 2 docvars.
       features
docs    end poverti in all it form everywher by 2030 erad
  text1   2       5  9   7  3    2         2  6    5    2
  text2   3       0 11   5  0    2         0  7    4    0
  text3   2       0  7   7  0    0         0  8    6    0
  text4   0       0  6   8  0    0         0  9    8    0
  text5   1       0  5   9  0    3         1  0    0    0
  text6   1       0  3   5  0    0         0  8    6    0
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 862 more features ]
```

Top features



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Filter by frequency

Very rare words and very frequent words are unlikely to be helpful in discriminating between documents.

Frequency-filtering in R

```
sdg_dfm <- sdg_corpus %>%
           tokens() %>%
           dfm()

sdg_dfm
```

Frequency-filtering in R

```r
sdg_dfm_filtered <- sdg_corpus %>%
            tokens(remove_punct = TRUE) %>%
            dfm() %>%
            # Remove all words that appear fewer than 3 times in the corpus
            dfm_trim(min_termfreq = 3)

sdg_dfm_filtered
```

```
Document-feature matrix of: 17 documents, 323 features (70.86% sparse) and 2 docvars.
       features
docs     end poverty in all its forms everywhere by 2030 eradicate
  text1    2       5  9   7   3     2          2  6    5         1
  text2    3       0 11   5   0     2          0  7    4         0
  text3    2       0  7   7   0     0          0  8    6         0
  text4    0       0  6   8   0     0          0  9    8         0
  text5    1       0  5   9   0     3          1  0    0         0
  text6    1       0  3   5   0     0          0  8    6         0
[ reached max_ndoc ... 11 more documents, reached max_nfeat ... 313 more features ]
```

Feature Comparison

```r
dim(sdg_dfm)
```

```
[1]   17 1085
```

```r
dim(sdg_dfm_bigram)
```

```
[1]   17 4337
```

```r
dim(sdg_dfm_trigram)
```

```
[1]   17 8685
```

```r
dim(sdg_dfm_no_stop)
```

```
[1]   17 1034
```

```r
dim(sdg_dfm_stem)
```

```
[1]  17 872
```

```r
dim(sdg_dfm_filtered)
```

```
[1]  17 323
```

```r
sparsity(sdg_dfm)
```

```
[1] 0.8640824
```

```r
sparsity(sdg_dfm_bigram)
```

```
[1] 0.9046237
```

```r
sparsity(sdg_dfm_trigram)
```

```
[1] 0.9186359
```

```r
sparsity(sdg_dfm_no_stop)
```

```
[1] 0.876152
```

```r
sparsity(sdg_dfm_stem)
```

```
[1] 0.8436994
```

```r
sparsity(sdg_dfm_filtered)
```

```
[1] 0.7086141
```

- Feature selection matters! See Denny and Spirling, 2017
- Just seven (binary) preprocessing decisions leads to a total of $2^7 = 128$ possible feature matrices
- These selection decisions can have substantive implications for the inferences we draw from QTA

Choosing between representations

How should we select between these representations?

1. There is no single “best” dfm
2. The optimal representation of a corpus will depend on the particular research task
    - Would you want to remove stop words when trying to detect gendered hate speech?
    - Would you want to stem if you wanted to measure future-oriented language?
    - Would you want to discard rare words when calculating linguistic complexity?
3. We need to design ways of *validating* the representations we construct

Break

Dictionaries

Motivating Example

Are female politicians less aggressive than male politicians? (Hargrave and Blumenau, 2022)

A repeated claim in the qualitative literature on gender and politics is that male and female politicians have distinct styles. Many political observers argue that women are less aggressive in political debate than their male colleagues. Most of the evidence for these claims is taken from small-N classical content analysis studies. We will review this question by applying an existing sentiment dictionary to a large-N corpus of parliamentary texts.



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Motivating Example

Are female politicians less aggressive than male politicians? (Hargrave and Blumenau, 2022)

A repeated claim in the qualitative literature on gender and politics is that male and female politicians have distinct styles. Many political observers argue that women are less aggressive in political debate than their male colleagues. Most of the evidence for these claims is taken from small-N classical content analysis studies. We will review this question by applying an existing sentiment dictionary to a large-N corpus of parliamentary texts.

Motivating Example

How might we conceptualize “aggression” in the context of parliamentary debate?

> Use of aggressive or combative language, which might include criticisms or insults; language that suggests forceful action; or declamatory or adversarial language.

1. Theoretical conceptualization
   - Existing literature makes frequent reference to the importance of combative language in politics
2. Empirical exploration/discovery
   - We can read and watch parliamentary debates to assess the ways in which aggression manifests in politicians’ speeches

Hand-coding: “Classic” content analysis

- Key feature: use of “human” coders to implement a pre-defined coding scheme, by reading and coding texts
- Human decision-making is the central feature of coding decisions, not a computer or other mechanized tool
- Validity is usually the objective, rather than reliability
  - Validity: am I measuring what I am claiming to measure?
  - Reliability: am I able to reliably replicate my coding?
- Example: hand-coding sentences into pre-defined categories

Bridging Qualitative and Quantitative Text Analyses

Dictionaries represent a hybrid procedure that bridges qualitative approaches and fully-automated text-as-data approaches

- "Qualitative" since it involves identification of the concepts and associated keys/categories, and the textual features associated with each key/category
  - Dictionary construction involves a lot of contextual interpretation and qualitative judgment
- "Quantitative" because it involves applying an algorithm to large corpora and presenting statistical summaries of results
  - Perfect reliability because there is no human decision making as part of the text analysis procedure

Rationale for dictionaries

- Rather than count all words that occur, pre-define words as associated with specific meanings
- Two components:
  1. **key**: the label for the equivalence class for the concept or canonical term
  2. **values**: (multiple) terms or patterns that are declared equivalent occurences of the key class
- A better metaphor is really a **thesaurus**: a canoncial term or concept (the key) associated with equivalent synonyms (the values)

Key	**Values**
Dog	Dalmation, Labrador, Poodle, Pug
Computation	Data, Number, Computer, Simulation
Genetics	Gene, DNA, Inherit

Counting words

A dictionary is just a list of words ($m = 1, \ldots, M$) that is related to a common concept.

Aggression
stupid
dishonest
lier
idiot
ignorant
hate
fight
battle

Counting words

Applying a dictionary to a corpus of texts ($i = 1, \ldots, N$) simply requires counting the number of times each word occurs in each text and summing them.

If W_{im} is the number of times word m appears in text i and 0 otherwise, then the dictionary score for document i is:

$$t_i = \frac{\sum_{m=1}^{M} W_{im}}{N_i}$$

Or, the proportion of words in document i that appear in the dictionary.

Counting words

> “That statement is as barbaric as it is downright **stupid**; it is nothing more than an **ignorant**, cruel and deliberate misconception to hide behind.”

$$t_i = \frac{\sum_{m=1}^{M} W_{im}}{N_i} = \frac{1+1}{14} = 0.14$$

Counting *weighted* words

A slight development on this would be to assign each word in the dictionary a *weight* which reflects something about the importance of the word to the concept

Aggression	Weight
stupid	.6
dishonest	.2
lie	.5
idiot	.7
ignorant	.3
brutal	.4
violence	.5

- Weights are implicit in *all* dictionary approaches.
- Typically, all words are counted equally which implies a score of 1 for all words.
- This is not necessarily correct!

Counting *weighted* words

We can adjust the previous formula to incorporate the weights (s_m):

$$t_i = \frac{\sum_{m=1}^{M} s_m W_{im}}{N_i}$$

Why normalise by N_i?

Some texts will be longer than others and we do not want these texts to mechanically be assigned higher scores.

Counting *weighted* words

> "That statement is as barbaric as it is downright **stupid**; it is nothing more than an **ignorant**, cruel and deliberate misconception to hide behind."

$$t_i = \frac{\sum_{m=1}^{M} s_m W_{im}}{N_i} = \frac{(1 \cdot 0.6) + (1 \cdot 0.3)}{14} = 0.06$$

Weights or no weights?

Most applications of dictionary methods in social science applications use unweighted dictionaries.

Why learn this then?

1. The equal weighting assumption is not necessarily reasonable or effective.
2. The idea of assigning weights to words is something that will come up many times in future weeks.

Advantages of dictionaries: Many existing implementations

Advantages of dictionaries: Multi-lingual

Disadvantage: Off-the-Shelf Dictionaries and Context

Applying off-the-shelf dictionaries to new contexts can be problematic:

- Problem 1: **polysemes** – words that have multiple meanings
  - Loughran and McDonald classify sentiment for a corpus of 50,115 firm-year 10-K filings from 1994–2008
  - Almost three-fourths of the "negative" words in their dictionary were typically not negative in a financial context: e.g. *tax*, *cost*, *liability*, *foreign*, *vice*, etc
- Problem 2: Dictionaries often lack important words in a given context
  - e.g. negative financial words such as *felony*, *litigation*, *restated*, *misstatement*, and *unanticipated*
- Problem 3: Some dictionaries might do more to pick up the *topic* of a document than the *tone* of a document

Disdvantages of Dictionaries

> “That statement is as barbaric as it is downright **stupid**; it is nothing more than an **ignorant**, cruel and deliberate misconception to hide behind.”

> “Terrible acts of **brutality** and **violence** have been carried out against the Rohingya people.”

Application

Applying dictionaries in quanteda

```
library(quanteda)
aggression_texts <- read.csv("aggression_texts.csv")
aggression_words <- read.csv("aggression_words.csv")[,1]
```

1. `aggression_texts` is a data.frame which includes 10937 sentences from parliamentary speeches
2. `aggression_words` is a vector of 222 words from the an existing “Aggression” dictionary

Our goal is to use `aggression_words` to score the texts in `aggression_texts`.

Aggressive Words?

```
print(aggression_words)
```

```
 [1] "abhor*"        "abus*"         "abusiv*"       "accus*"
 [5] "afflict*"      "aggress*"      "aggressiv*"    "ambush*"
 [9] "anger*"        "angri*"        "angrier*"      "angry*"
[13] "annihilat*"    "annoy*"        "annoyanc*"     "antagoniz*"
[17] "argu*"         "argument*"     "army*"         "arrow*"
[21] "assault*"      "attack*"       "aveng*"        "ax"
[25] "axe"           "axes"          "battl*"        "beak*"
[29] "beat*"         "beaten*"       "betray*"       "blade*"
[33] "blam*"         "bloody*"       "bother*"       "brawl*"
[37] "break*"        "brok*"         "broken*"       "brutal*"
[41] "cannon*"       "chid*"         "combat*"       "complain*"
[45] "conflict*"     "condemn*"      "controversy*"  "critic*"
[49] "cruel*"        "crush*"        "cut"           "cuts"
[53] "cutt*"         "damag*"        "decei*"        "defeat*"
[57] "degrad*"       "demolish*"     "depriv*"       "derid*"
[61] "despis*"       "destroy*"      "destruct*"     "destructiv*"
[65] "detest*"       "disagre*"      "disagreement*" "disapprov*"
[69] "discontent*"   "dislik*"       "disput*"       "disturb*"
[73] "doubt*"        "enemi*"        "enemy*"        "enrag*"
[77] "exasperat*"    "controversial*" "critique"     "disparag*"
[81] "irritable"     "exploit*"      "exterminat*"   "feud*"
[85] "fierc*"        "fight*"        "fought*"       "furiou*"
```

The $*$ character will pick up any token which begins with the relevant string.

I.e. accus$*$ ➡️ accuse, accuses, accused, etc.

Applying Dictionaries in Quanteda

```r
# First we convert the texts to a corpus object:
aggression_corpus <- corpus(aggression_texts, text_field = "texts")

# Then we tokenize the texts and create a dfm:
aggression_tokens <- tokens(aggression_corpus)
aggression_dfm <- dfm(aggression_tokens)

# We use the aggression words to create a dictionary object:
aggression_dictionary <- dictionary(list(aggression = aggression_words))

# Finally, we apply the dictionary to the dfm using the dfm_lookup function:
aggression_dfm_dictionary <- dfm_lookup(aggression_dfm,
                                        dictionary = aggression_dictionary)
```

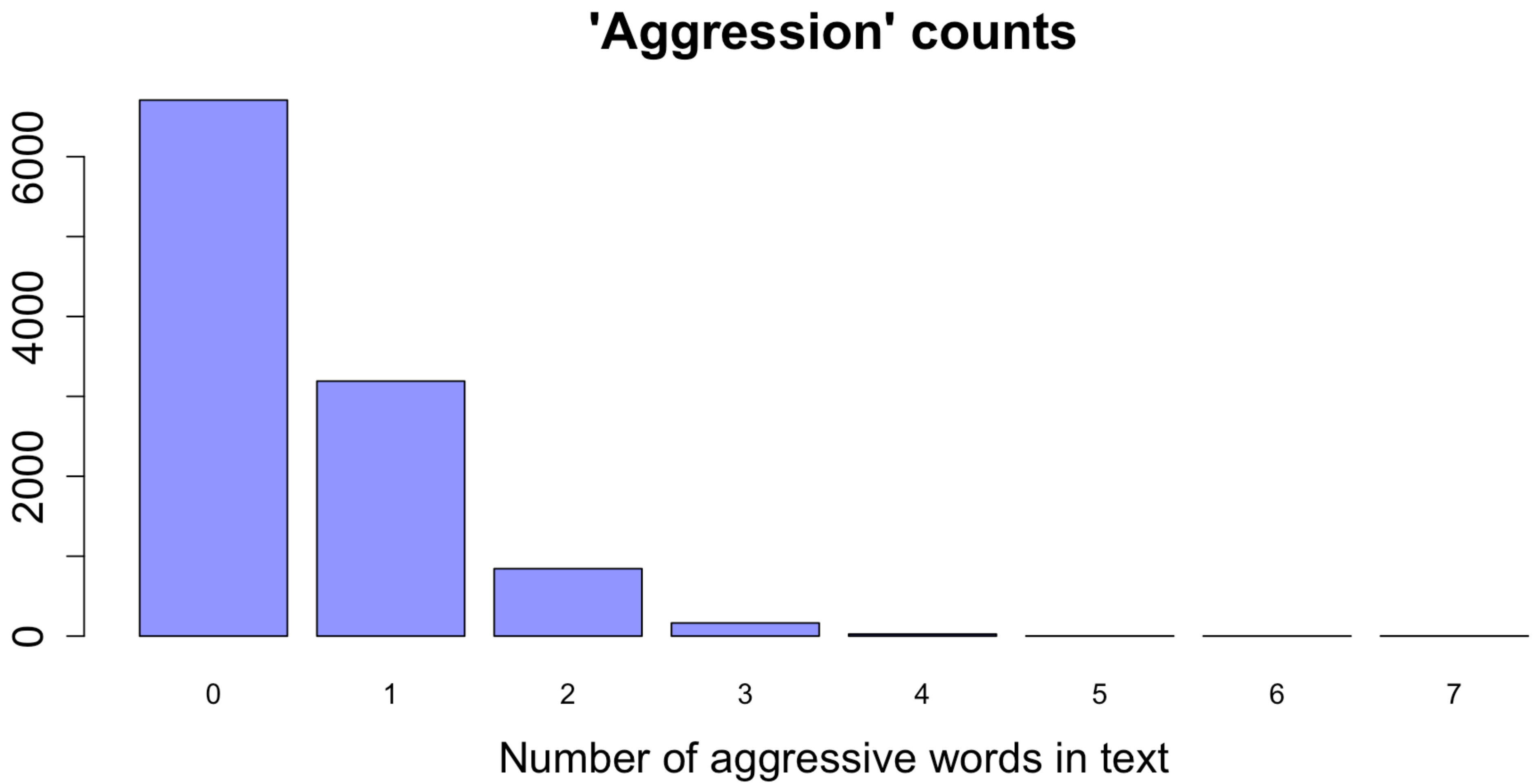
Applying Dictionaries in Quanteda

```
print(aggression_dfm_dictionary)
```

```
Document-feature matrix of: 10,937 documents, 1 feature (79.05% sparse) and 1 docvar.
       features
docs    aggression
  text1          0
  text2          0
  text3          0
  text4          0
  text5          1
  text6          1
[ reached max_ndoc ... 10,931 more documents ]
```

`aggression_dfm` is a document-feature matrix, where the only "feature" is the dictionary counts

Applying Dictionaries in Quanteda



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Applying Dictionaries in Quanteda

Finally, we can calculate the score by dividing the dictionary counts by the number of words in each text:

```
aggression_texts$proportions <- as.numeric(aggression_dfm_dictionary[,1]) /
  ntoken(aggression_corpus)
```

```
summary(aggression_texts$proportions)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.00000 0.00000 0.00000 0.00811 0.00000 0.19048 
```

```
hist(aggression_texts$proportions)
```

Validation

Validation tests

- Applying dictionaries outside the domain for which they were developed can lead to errors.
- One way of assessing the seriousness of these errors is to conduct **validation tests**
- There are many forms of these tests!
- All share a core idea: are the texts that are flagged by the dictionary more representative of the relevant concept than other texts?

Types of validation

There are many approaches to assessing validity of a measure, m_1 , for a target concept, μ_1 :

THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Human Judgement as a "Gold Standard"

- Comparison to human judgements of a target concept, μ, are often thought to be the "gold standard" of validation
- This is based on the (often implicit) assumption that real people can accurately identify and label examples of a given concept ("you know it when you see it")
- This assumption may not be met due to…
  - Misinterpretation
  - Poor/unclear conceptualisation
  - Lack of coder training
  - Etc
- The caratage of the gold standard will therefore vary across applications

Face validity (1)

Intuition: Does our measure of aggression vary in sensible ways?

In this case, one obvious test is whether MPs speeches are more aggressive during Prime Minister's Questions (PMQs).

Face validity (1)

```r
str(aggression_texts)
```

```
'data.frame':	10937 obs. of  4 variables:
 $ texts      : chr  "Is it not more important to work hard to open up trade between eastern and western Europe than to allow the Eur"| __truncated__ "Also, the Bill will consider aspects of the procedures applying to boards of inquiry." "On that measure, NHS provision per head of population in Cornwall is about half the national average." "Making it a criminal offence would help to make it clear that forced marriage is completely and utterly unacceptable." ...
 $ human      : logi  NA NA NA NA NA NA ...
 $ debate_type: chr  "legislation" "question_time" "question_time" "question_time" ...
 $ proportions: num  0 0 0 0 0.0233 ...
```

```r
table(aggression_texts$debate_type)
```

```

              legislation              opposition_day prime_ministers_questions 
                     4799                         965                      1400 
            question_time 
                     3773 
```

Face validity (1)

```
library(tidyverse) # Load libraries

aggression_texts %>% # Pipe the aggression texts object
  group_by(debate_type) %>% # Group data by the debate_type variable
  summarise(mean_dictionary = mean(proportions)) # Calculate the mean dictionary score for each type
```

```
# A tibble: 4 × 2
  debate_type               mean_dictionary
  <chr>                               <dbl>
1 legislation                       0.00656
2 opposition_day                    0.00685
3 prime_ministers_questions         0.0171
4 question_time                     0.00706
```

Face validity (1)

```r
summary(lm(proportions ~ debate_type, data = aggression_texts))
```

```

Call:
lm(formula = proportions ~ debate_type, data = aggression_texts)

Residuals:
      Min        1Q    Median        3Q       Max
-0.017121 -0.007060 -0.006559 -0.006559  0.173355

Coefficients:
                                      Estimate Std. Error t value Pr(>|t|)
(Intercept)                          0.0065592  0.0002597  25.261   <2e-16 ***
debate_typeopposition_day            0.0002897  0.0006346   0.456    0.648
debate_typeprime_ministers_questions 0.0105622  0.0005464  19.331   <2e-16 ***
debate_typequestion_time             0.0005009  0.0003914   1.280    0.201
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.01799 on 10933 degrees of freedom
Multiple R-squared:  0.03569,	Adjusted R-squared:  0.03542
F-statistic: 134.9 on 3 and 10933 DF,  p-value: < 2.2e-16
```

There is clear evidence that PMQ debates tend to have higher levels of aggressive language than other debates.

Face validity (2)

How does this approach perform? Let's look at the top-scoring sentences:

	score	text
text3998	0.19	I fully appreciate that it is the Opposition's job to oppose, but there are times when opposition is destructive.
text7416	0.18	We unequivocally condemn Hamas's dreadful and murderous rocket attacks and defend Israel's right to defend itself.
text2941	0.14	They were asking ridiculous prices, because they had the sole remedy for a complaint, so could exploit that situation.
text106	0.13	Terrible acts of brutality and violence have been carried out against the Rohingya people.
text144	0.13	The motion condemns the early release scheme for those who have assaulted police officers.

While some seem reasonable, others indicate that we are picking up topic rather than tone.

Comparison to Human Judgement

- The `aggression_texts` data.frame includes a variable, `human`, which includes the results of a validation exercise.

```
str(aggression_texts)
```

```
'data.frame':	10937 obs. of  4 variables:
 $ texts      : chr  "Is it not more important to work hard to open up trade between eastern and western Europe than to allow the Eur"| __truncated__ "Also, the Bill will consider aspects of the procedures applying to boards of inquiry." "On that measure, NHS provision per head of population in Cornwall is about half the national average." "Making it a criminal offence would help to make it clear that forced marriage is completely and utterly unacceptable." ...
 $ human      : logi  NA NA NA NA NA NA ...
 $ debate_type: chr  "legislation" "question_time" "question_time" "question_time" ...
 $ proportions: num  0 0 0 0 0.0233 ...
```

```
table(dictionary = aggression_texts$proportions > 0,
      human = aggression_texts$human)
```

```
          human
dictionary FALSE TRUE
     FALSE   674  124
     TRUE     75  127
```

```
(127 + 674)/1000
```

```
[1] 0.801
```

- Is this good?

Paired Comparisons versus Single Ratings

Which of these questions is easier?

1. Is this sentence aggressive?
    - "I regard it as unbelievable that the minister has said that, when it is clearly wrong."
2. Which of these sentences is more aggressive?
    - "I regard it as unbelievable that the minister has said that, when it is clearly wrong."
    - "I also welcome the fact that the Bill will encourage more young people to take advantage of the programme."

Paired comparisons tend to give more useful and reliable information than single ratings.

Paired Comparisons versus Single Ratings

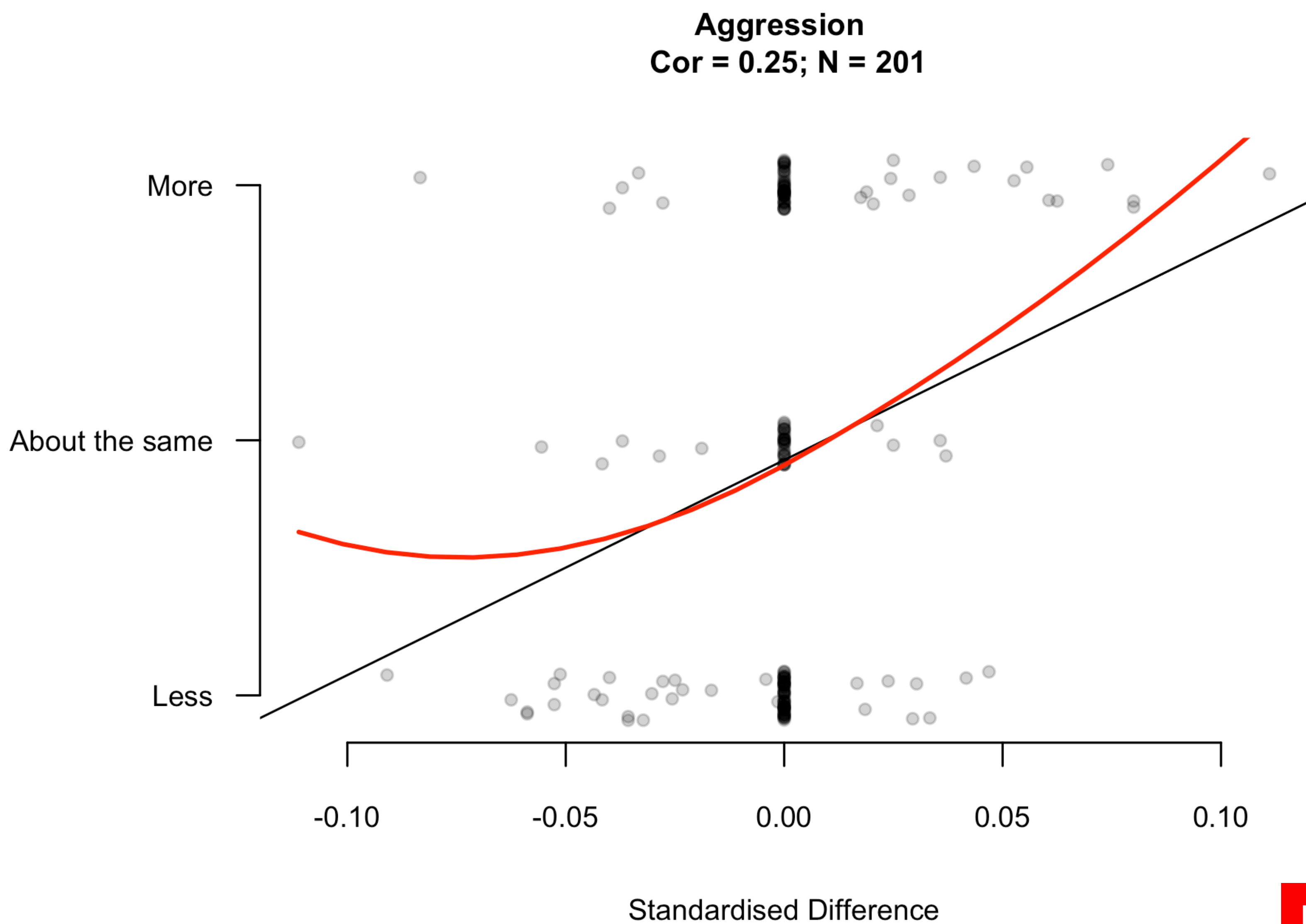
1. Apply 7 basic QTA measures (including 6 dictionaries) to 8 million sentences
   - Aggression
   - Positive Emotion
   - Negative Emotion
   - Fact
   - Anecdote
   - Complexity
   - Repetition
2. Score each sentence using uniform word weights
3. Present pairs of sentences to human coders and ask them to select which sentence is most representative of a certain concept

Validation Measure

Does the difference in sentence-level dictionary scores predict human judgements?

- Sample pairs of sentences from the corpus
  - Score each pair as $\mathrm{Diff}_i = t_{2i} - t_{1i}$
- Randomly present to human coders, code (Y_i) whether:
  - Sentence one is more <style> (1)
  - About the same (0)
  - Sentence two is more <style> (-1)
- Calculate the relationship between human coding and dictionaries by:
  - $Y_i = \alpha + \beta \mathrm{Diff}_i$
  - $\mathrm{Cor}(Y_i, \mathrm{Diff}_i)$
- Repeat for each dictionary

Validation Results



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Interpretation

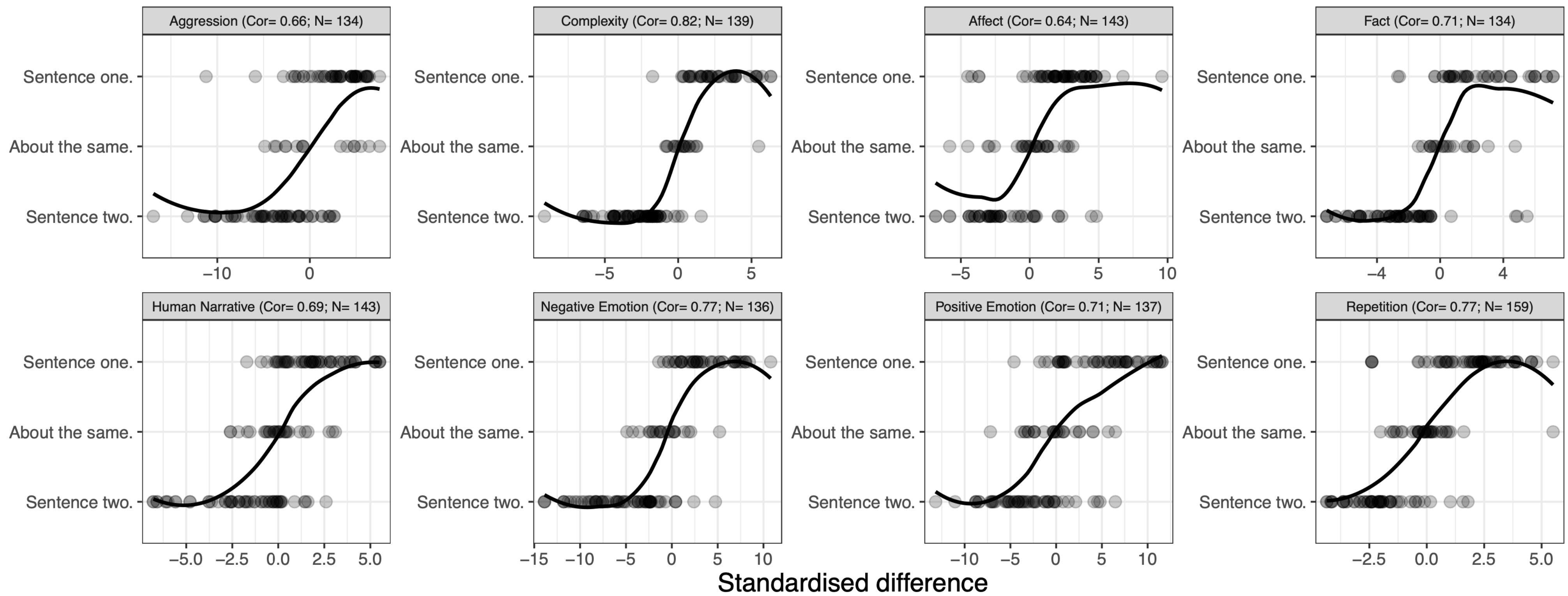
Aggression tends to manifest very differently in parliamentary speech than in other contexts!

In the paper, H&B develop a more sophisticated approach to measuring aggression (and other styles):

1. Take an off-the-shelf dictionary of aggressive words
2. Use word-embeddings to…
    a. …expand the initial dictionary to include words that are relevant to parliamentary speeches
    b. …upweight words that are used in a similar way in parliamentary speech
    c. …downweight words that are not typically used in a similar way in parliamentary speech
3. Score speeches according to these modified word lists

More on this approach on Thursday.

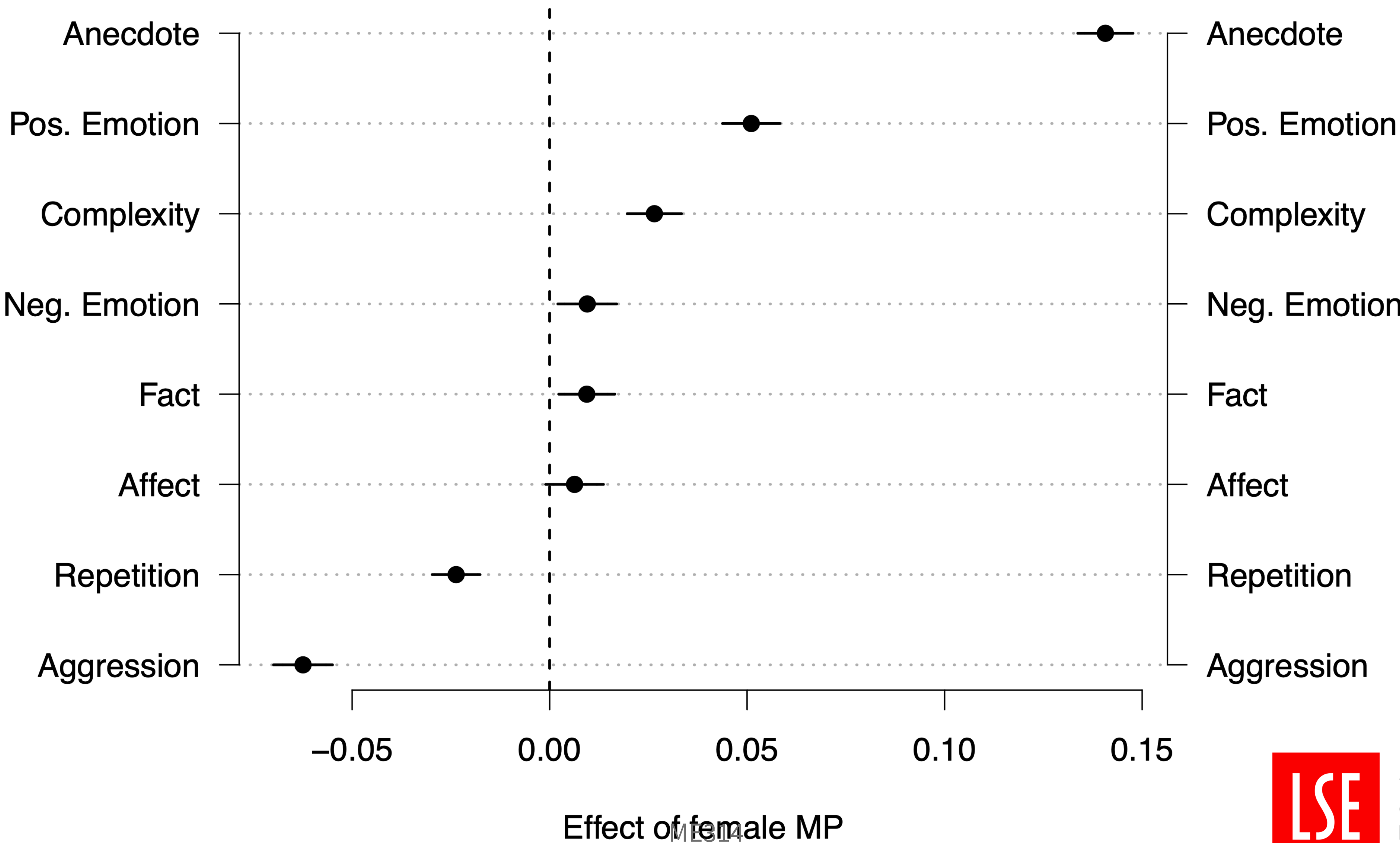
Word-embedding Results



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

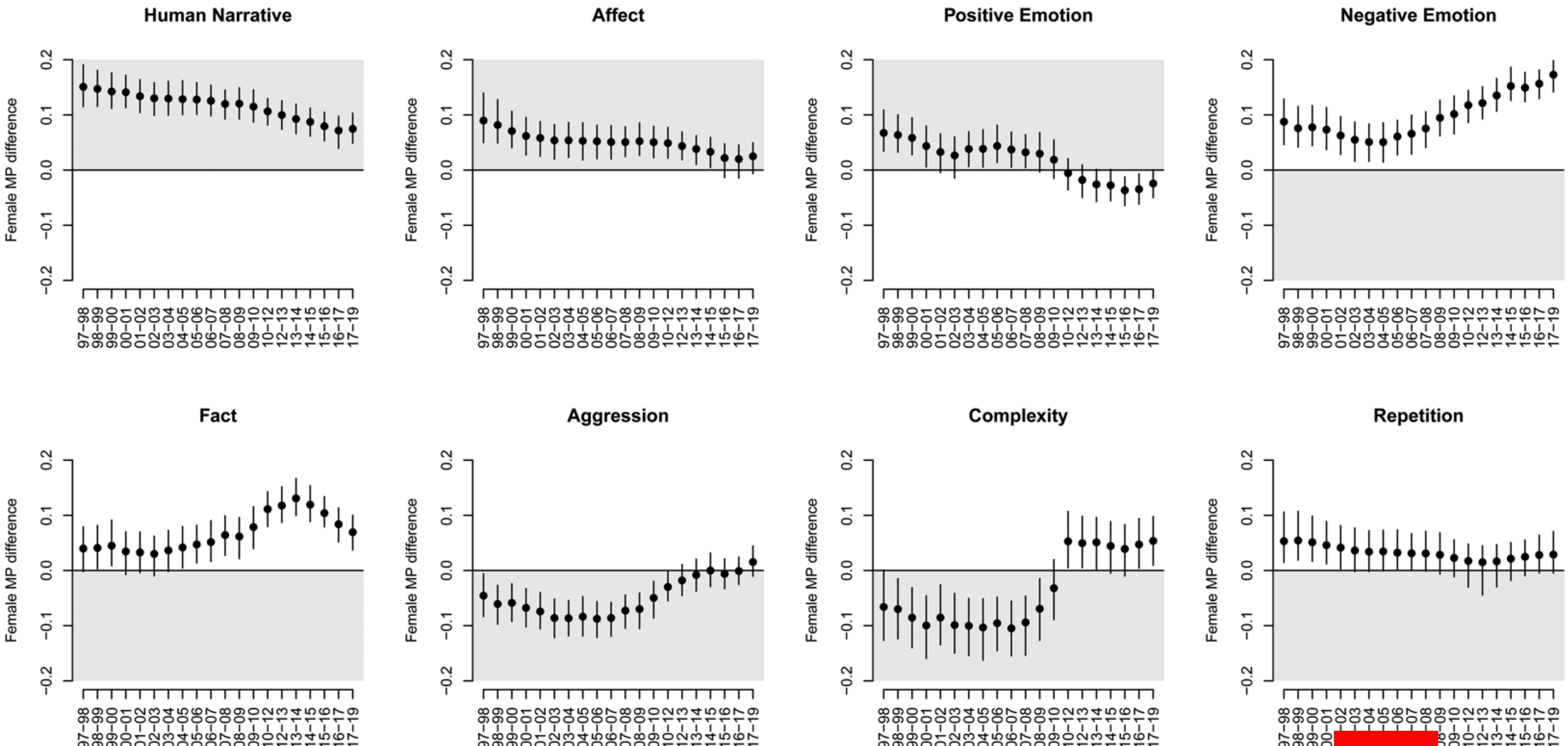
Are women less aggressive than men?

Let's believe for a second that the validation strategy worked.



THE LONDON SCHOOL OF ECONOMICS AND POLITICAL SCIENCE

Have political styles changed over time?



Full paper here.

Conclusion

Summing Up

- Quantitative Text Analysis allows us to address a wide variety of important research questions
- There is no one right way to represent text for all research questions.
- The representation we choose can be consequential for the results we present
- Dictionaries are fast, easy-to-apply, methods with many pre-existing implementations
- Validation is critical to any quantitative text application
- The validity of a dictionary will be sensitive to the contexts in which it is developed and applied