Quantitative Text Analysis

School of Public Policy
Oregon State University

Christopher Zorn
zorn@psu.edu

January 19, 2023

Workshop things:

- **"Quantitative Text Analysis"**
- Logistics: Meet January 19, 2023 from ∼9:00 a.m. until ∼4:30 p.m. PT in Bexell Hall room 120
- All course materials, code, data, examples, etc. are on the workshop Github repository, at:

  `https://github.com/PrisonRodeo/OSU-Text`

- Software: R (see the repository for more...)
  - Code is available here
  - Commands will typically require loading the packages at the beginning of the code file
- Me: Chris Zorn, (mostly) from Political Science at Penn State

Le Menu

(0. An R Introduction)

1. Text As Data
2. Working With Text
3. “Bag of Words” Approaches
4. NLP: Transformers, LLMs, etc.

A Little Introduction to R

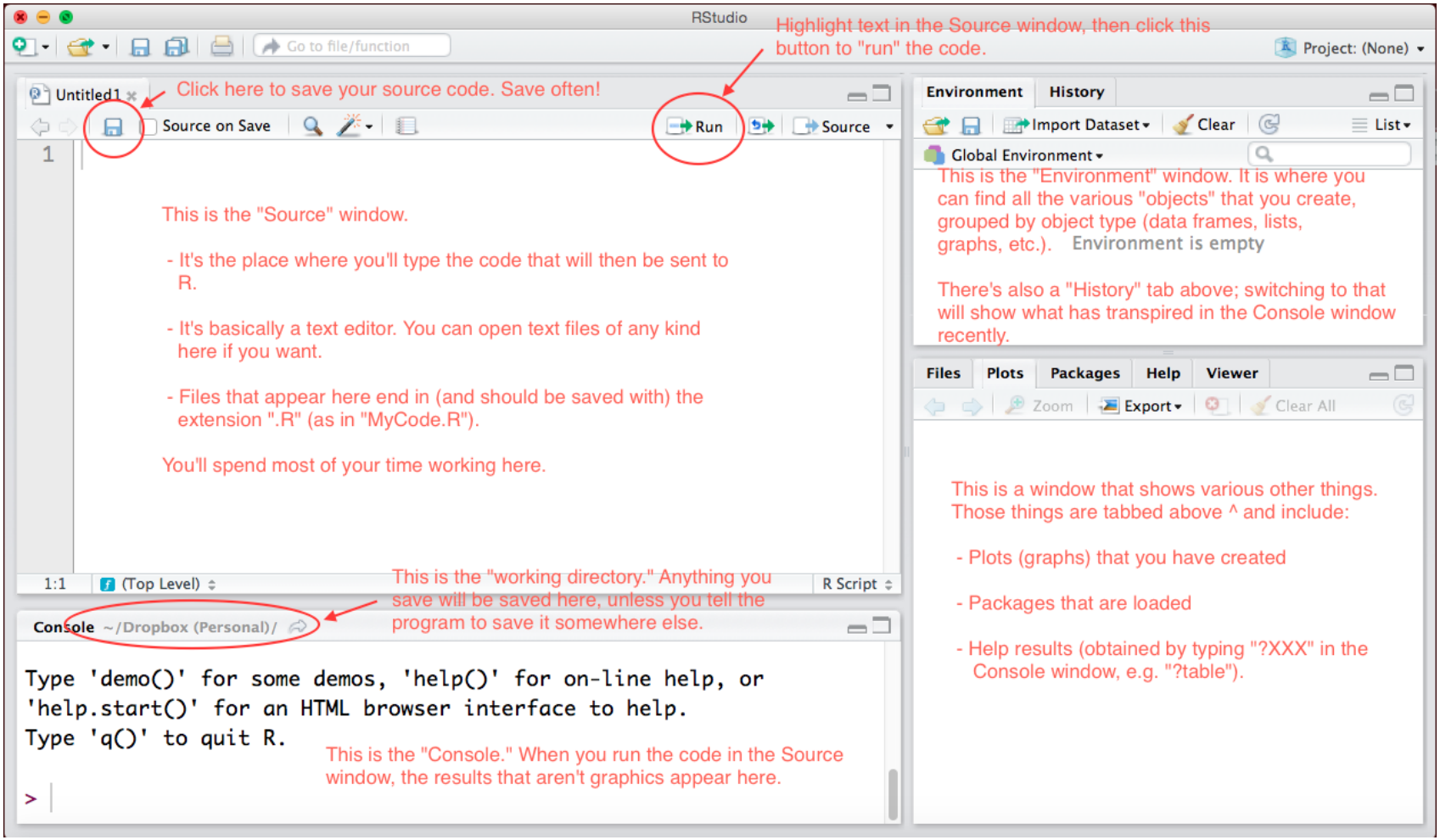
R (and RStudio)

- R
  - "R is a free software environment for statistical computing and graphics."
  - The R Project: https://www.r-project.org/
  - Comprehensive R Archive Network (CRAN): https://cran.r-project.org/
- RStudio / "Posit"
  - A free, open-source GUI for R
  - Website: https://posit.co/
  - Also available in the Cloud (https://posit.cloud/)

R:

- Is an **object-oriented** language
- Is made up of:
  - Objects
  - Functions
  - Classes (of objects and functions)
- Is **Turing complete** + **regex-capable**
- Is **modular**
  - User-created **packages**
  - Organized into **task views** (https://cran.r-project.org/web/views/)
- Runs on UNIX/Linux/MacOS/Windows

RStudio (annotated)



The "assignment operator":

```
CZ <- 54
AJ <- "The boss."
```

Object types:

```
# Vector:
A <- c(1,2,3,4,5,6)

# Matrix:
B <- matrix(data=c(1,2,3,4,5,6),nrow=2,ncol=3)

# Character:
C <- "Pinot Noir"

# "Data frame":

D <- data.frame(ID=1:100,
                Foo=rnorm(100),
                Bar=rep(C,100))
```

Miscellaneous Good R Practices

In *general*:

- Load all packages early in your code:

```
>P <- c("devtools","readr","haven","plyr","dplyr",
     "statmod","lubridate","stringr","MASS","httr")
> for (i in 1:length(P)) {
      ifelse(!require(P[i], character.only=TRUE),install.packages(P[i]), print(":"))
      library(P[i],character.only=TRUE)
      }
> rm(P,i)
```

- Write lots of comments:

```
> fit <- lm(y~x1+x2,data=DF) # This is a linear regression!
```

- Pull data from a reliable/archival source each time:

```
> COVID.data<-read_csv("https://github.com/nytimes/covid-19-data/raw/master/us-counties-2023.csv")
```

Text As Data

Analyzing Text: Goals

Humans:

- Good at: Meaning, subtlety (irony, sarcasm, subtle negation, etc.), context, tone, etc.
- Bad at: Doing things quickly and consistently.

Computers:

- Good at: Doing things quickly and consistently.
- Bad at: Meaning, subtlety (irony, sarcasm, subtle negation, etc.), context, tone, etc.

What Can (and Can't) We Do With Text?

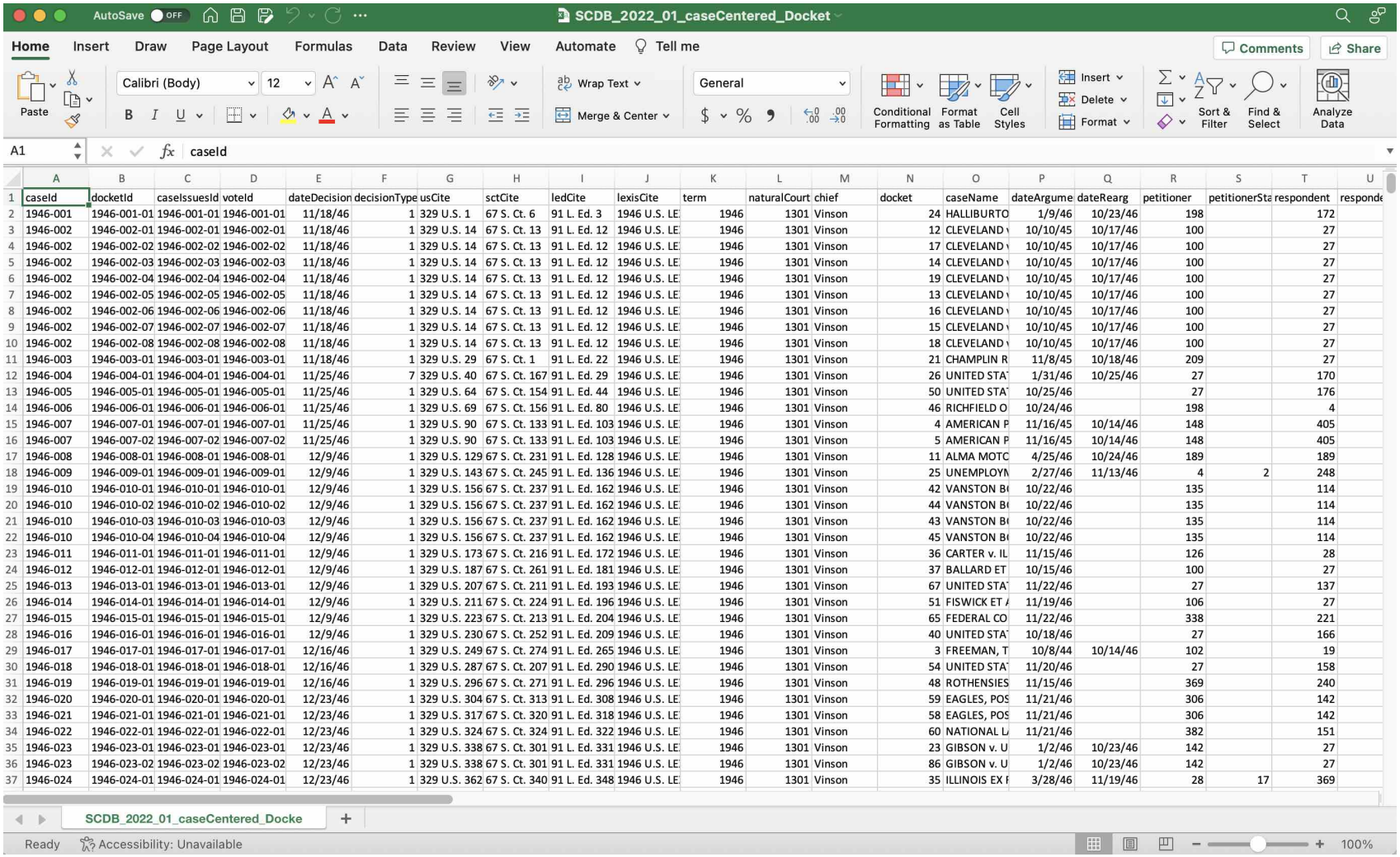
Can: "Improved reading" (Grimmer)

- *Compare*, *organize*, and *classify* text
- *Detect* and *measure* social phenomena
- *Amplify* resources and *augment* people

Can't:

- Develop a comprehensive statistical model of language
- Replace the need to *read*

Data, As We Know It



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	caseId	docketId	caseIssuesId	voteId	dateDecision	decisionType	usCite	sctCite	ledCite	lexisCite	term	naturalCourt	chief	docket	caseName	dateArgume	dateRearg	petitioner	petitionerSta	respondent	respondé
2	1946-001	1946-001-01	1946-001-01	1946-001-01	11/18/46	1	329 U.S. 1	67 S. Ct. 6	91 L. Ed. 3	1946 U.S. LE	1946	1301	Vinson	24	HALLIBURTO	1/9/46	10/23/46	198		172	
3	1946-002	1946-002-01	1946-002-01	1946-002-01	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	12	CLEVELAND v	10/10/45	10/17/46	100		27	
4	1946-002	1946-002-02	1946-002-02	1946-002-02	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	17	CLEVELAND v	10/10/45	10/17/46	100		27	
5	1946-002	1946-002-03	1946-002-03	1946-002-03	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	14	CLEVELAND v	10/10/45	10/17/46	100		27	
6	1946-002	1946-002-04	1946-002-04	1946-002-04	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	19	CLEVELAND v	10/10/45	10/17/46	100		27	
7	1946-002	1946-002-05	1946-002-05	1946-002-05	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	13	CLEVELAND v	10/10/45	10/17/46	100		27	
8	1946-002	1946-002-06	1946-002-06	1946-002-06	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	16	CLEVELAND v	10/10/45	10/17/46	100		27	
9	1946-002	1946-002-07	1946-002-07	1946-002-07	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	15	CLEVELAND v	10/10/45	10/17/46	100		27	
10	1946-002	1946-002-08	1946-002-08	1946-002-08	11/18/46	1	329 U.S. 14	67 S. Ct. 13	91 L. Ed. 12	1946 U.S. LE	1946	1301	Vinson	18	CLEVELAND v	10/10/45	10/17/46	100		27	
11	1946-003	1946-003-01	1946-003-01	1946-003-01	11/18/46	1	329 U.S. 29	67 S. Ct. 1	91 L. Ed. 22	1946 U.S. LE	1946	1301	Vinson	21	CHAMPLIN R	11/8/45	10/18/46	209		27	
12	1946-004	1946-004-01	1946-004-01	1946-004-01	11/25/46	7	329 U.S. 40	67 S. Ct. 167	91 L. Ed. 29	1946 U.S. LE	1946	1301	Vinson	26	UNITED STA	1/31/46	10/25/46	27		170	
13	1946-005	1946-005-01	1946-005-01	1946-005-01	11/25/46	1	329 U.S. 64	67 S. Ct. 154	91 L. Ed. 44	1946 U.S. LE	1946	1301	Vinson	50	UNITED STA	10/25/46		27		176	
14	1946-006	1946-006-01	1946-006-01	1946-006-01	11/25/46	1	329 U.S. 69	67 S. Ct. 156	91 L. Ed. 80	1946 U.S. LE	1946	1301	Vinson	46	RICHFIELD O	10/24/46		198		4	
15	1946-007	1946-007-01	1946-007-01	1946-007-01	11/25/46	1	329 U.S. 90	67 S. Ct. 133	91 L. Ed. 103	1946 U.S. LE	1946	1301	Vinson	4	AMERICAN P	11/16/45	10/14/46	148		405	
16	1946-007	1946-007-02	1946-007-02	1946-007-02	11/25/46	1	329 U.S. 90	67 S. Ct. 133	91 L. Ed. 103	1946 U.S. LE	1946	1301	Vinson	5	AMERICAN P	11/16/45	10/14/46	148		405	
17	1946-008	1946-008-01	1946-008-01	1946-008-01	12/9/46	1	329 U.S. 129	67 S. Ct. 231	91 L. Ed. 128	1946 U.S. LE	1946	1301	Vinson	11	ALMA MOTC	4/25/46	10/24/46	189		189	
18	1946-009	1946-009-01	1946-009-01	1946-009-01	12/9/46	1	329 U.S. 143	67 S. Ct. 245	91 L. Ed. 136	1946 U.S. LE	1946	1301	Vinson	25	UNEMPLOYM	2/27/46	11/13/46	4	2	248	
19	1946-010	1946-010-01	1946-010-01	1946-010-01	12/9/46	1	329 U.S. 156	67 S. Ct. 237	91 L. Ed. 162	1946 U.S. LE	1946	1301	Vinson	42	VANSTON B	10/22/46		135		114	
20	1946-010	1946-010-02	1946-010-02	1946-010-02	12/9/46	1	329 U.S. 156	67 S. Ct. 237	91 L. Ed. 162	1946 U.S. LE	1946	1301	Vinson	44	VANSTON B	10/22/46		135		114	
21	1946-010	1946-010-03	1946-010-03	1946-010-03	12/9/46	1	329 U.S. 156	67 S. Ct. 237	91 L. Ed. 162	1946 U.S. LE	1946	1301	Vinson	43	VANSTON B	10/22/46		135		114	
22	1946-010	1946-010-04	1946-010-04	1946-010-04	12/9/46	1	329 U.S. 156	67 S. Ct. 237	91 L. Ed. 162	1946 U.S. LE	1946	1301	Vinson	45	VANSTON B	10/22/46		135		114	
23	1946-011	1946-011-01	1946-011-01	1946-011-01	12/9/46	1	329 U.S. 173	67 S. Ct. 216	91 L. Ed. 172	1946 U.S. LE	1946	1301	Vinson	36	CARTER v. IL	11/15/46		126		28	
24	1946-012	1946-012-01	1946-012-01	1946-012-01	12/9/46	1	329 U.S. 187	67 S. Ct. 261	91 L. Ed. 181	1946 U.S. LE	1946	1301	Vinson	37	BALLARD ET	10/15/46		100		27	
25	1946-013	1946-013-01	1946-013-01	1946-013-01	12/9/46	1	329 U.S. 207	67 S. Ct. 211	91 L. Ed. 193	1946 U.S. LE	1946	1301	Vinson	67	UNITED STA	11/22/46		27		137	
26	1946-014	1946-014-01	1946-014-01	1946-014-01	12/9/46	1	329 U.S. 211	67 S. Ct. 224	91 L. Ed. 196	1946 U.S. LE	1946	1301	Vinson	51	FISWICK ET A	11/19/46		106		27	
27	1946-015	1946-015-01	1946-015-01	1946-015-01	12/9/46	1	329 U.S. 223	67 S. Ct. 213	91 L. Ed. 204	1946 U.S. LE	1946	1301	Vinson	65	FEDERAL CO	11/22/46		338		221	
28	1946-016	1946-016-01	1946-016-01	1946-016-01	12/9/46	1	329 U.S. 230	67 S. Ct. 252	91 L. Ed. 209	1946 U.S. LE	1946	1301	Vinson	40	UNITED STA	10/18/46		27		166	
29	1946-017	1946-017-01	1946-017-01	1946-017-01	12/16/46	1	329 U.S. 249	67 S. Ct. 274	91 L. Ed. 265	1946 U.S. LE	1946	1301	Vinson	3	FREEMAN, T	10/8/44	10/14/46	102		19	
30	1946-018	1946-018-01	1946-018-01	1946-018-01	12/16/46	1	329 U.S. 287	67 S. Ct. 207	91 L. Ed. 290	1946 U.S. LE	1946	1301	Vinson	54	UNITED STA	11/20/46		27		158	
31	1946-019	1946-019-01	1946-019-01	1946-019-01	12/16/46	1	329 U.S. 296	67 S. Ct. 271	91 L. Ed. 296	1946 U.S. LE	1946	1301	Vinson	48	ROTHENSIES	11/15/46		369		240	
32	1946-020	1946-020-01	1946-020-01	1946-020-01	12/23/46	1	329 U.S. 304	67 S. Ct. 313	91 L. Ed. 308	1946 U.S. LE	1946	1301	Vinson	59	EAGLES, POS	11/21/46		306		142	
33	1946-021	1946-021-01	1946-021-01	1946-021-01	12/23/46	1	329 U.S. 317	67 S. Ct. 320	91 L. Ed. 318	1946 U.S. LE	1946	1301	Vinson	58	EAGLES, POS	11/21/46		306		142	
34	1946-022	1946-022-01	1946-022-01	1946-022-01	12/23/46	1	329 U.S. 324	67 S. Ct. 324	91 L. Ed. 322	1946 U.S. LE	1946	1301	Vinson	60	NATIONAL L	11/21/46		382		151	
35	1946-023	1946-023-01	1946-023-01	1946-023-01	12/23/46	1	329 U.S. 338	67 S. Ct. 301	91 L. Ed. 331	1946 U.S. LE	1946	1301	Vinson	23	GIBSON v. U	1/2/46	10/23/46	142		27	
36	1946-023	1946-023-02	1946-023-02	1946-023-02	12/23/46	1	329 U.S. 338	67 S. Ct. 301	91 L. Ed. 331	1946 U.S. LE	1946	1301	Vinson	86	GIBSON v. U	1/2/46	10/23/46	142		27	
37	1946-024	1946-024-01	1946-024-01	1946-024-01	12/23/46	1	329 U.S. 362	67 S. Ct. 340	91 L. Ed. 348	1946 U.S. LE	1946	1301	Vinson	35	ILLINOIS EX F	3/28/46	11/19/46	28	17	369	

SUPREME COURT OF THE UNITED STATES

No. 19–1392

THOMAS E. DOBBS, STATE HEALTH OFFICER OF THE MISSISSIPPI DEPARTMENT OF HEALTH, ET AL., PETITIONERS *v.* JACKSON WOMEN'S HEALTH ORGANIZATION, ET AL.

ON WRIT OF CERTIORARI TO THE UNITED STATES COURT OF APPEALS FOR THE FIFTH CIRCUIT

[June 24, 2022]

JUSTICE BREYER, JUSTICE SOTOMAYOR, and JUSTICE KAGAN, dissenting.

For half a century, *Roe* v. *Wade*, 410 U. S. 113 (1973), and *Planned Parenthood of Southeastern Pa.* v. *Casey*, 505 U. S. 833 (1992), have protected the liberty and equality of women. *Roe* held, and *Casey* reaffirmed, that the Constitution safeguards a woman's right to decide for herself whether to bear a child. *Roe* held, and *Casey* reaffirmed, that in the first stages of pregnancy, the government could not make that choice for women. The government could not control a woman's body or the course of a woman's life: It could not determine what the woman's future would be. See *Casey*, 505 U. S., at 853; *Gonzales* v. *Carhart*, 550 U. S. 124, 171–172 (2007) (Ginsburg, J., dissenting). Respecting a woman as an autonomous being, and granting her full equality, meant giving her substantial choice over this most personal and most consequential of all life decisions.

Roe and *Casey* well understood the difficulty and divisiveness of the abortion issue. The Court knew that Americans hold profoundly different views about the "moral[ity]" of "terminating a pregnancy, even in its earliest stage." *Casey*, 505 U. S., at 850. And the Court recognized that "the

Text: Where?

Text sources:

- The Web...
  - Social media, web pages, blogs, etc.
  - Native-digital documents
  - Archives, collections, etc.
- Audio / video transcripts
- Digitized Documents
- Surveys (open-ended questions)
- "Data Exhaust"

Text: How?

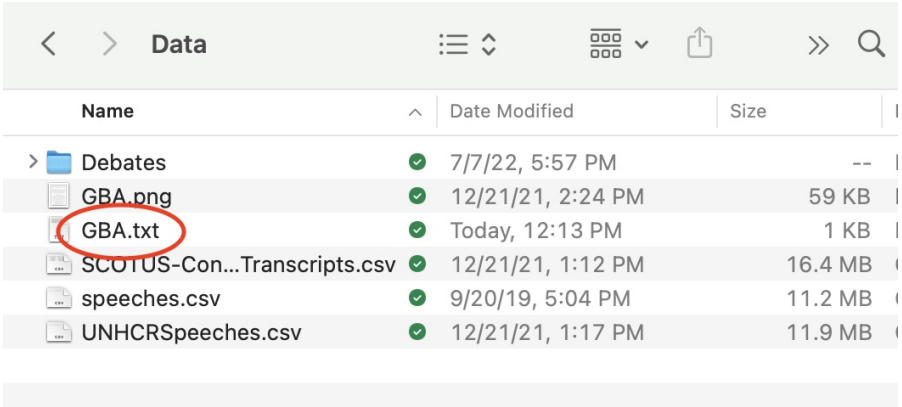
Text formats:

- Machine-encoded
  - Plain-text
  - PDFs, etc.
- Graphical / images $\rightarrow$ "OCR"

Text access:

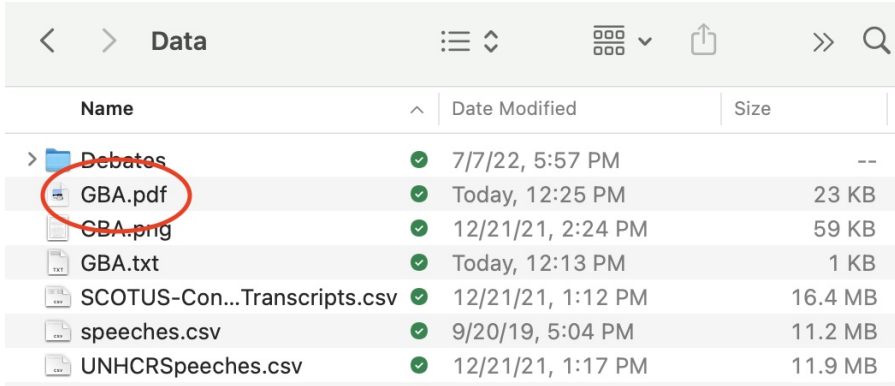
- Static URLs / FTP / etc.
- Dynamic URLs (PHP, etc.)
- Application programming interfaces (APIs)

"Local" Plain-Text




```
> GBA<-read_file("~/Dropbox (Personal)/OSU Workshop/Data/GBA.txt")
> GBA
[1] "Four score and seven years ago our fathers brought forth on this continent, a new nation,
conceived in Liberty, and dedicated to the proposition that all men are created equal.\n\nNow
we are engaged in a great civil war, testing whether that nation, or any nation so conceived
and dedicated, can long endure. We are met on a great battle-field of that war. We have come
to dedicate a portion of that field, as a final resting place for those who here gave their lives
that that nation might live. It is altogether fitting and proper that we should do this.\n\nBut, in
a larger sense, we can not dedicate -- we can not consecrate -- we can not hallow -- this
ground. The brave men, living and dead, who struggled here, have consecrated it, far above
our poor power to add or detract. The world will little note, nor long remember what we say here,
but it can never forget what they did here. It is for us the living, rather, to be dedicated here to
the unfinished work which they who fought here have thus far so nobly advanced. It is rather for
us to be here dedicated to the great task remaining before us -- that from these honored dead
we take increased devotion to that cause for which they gave the last full measure of devotion
-- that we here highly resolve that these dead shall not have died in vain -- that this nation, under
God, shall have a new birth of freedom -- and that government of the people, by the people, for
the people, shall not perish from the earth.\n"
```

"Local" PDF




```
> GBA.2<-pdf_text("~/Dropbox (Personal)/OSU Workshop/Data/GBA.pdf")
> GBA.2
[1] "Gettysburg address delivered at Gettysburg Pa. Nov. 19th, 1863. [n. p. n. d.].\nGettysburg
Address\n\nDelivered at Gettysburg, Pa.\n\nNov. 19th 1863.\n\n"Four score and seven years
ago our fathers brought forth on this continent a new nation, conceived\nin liberty, and dedicated
to the proposition that all men are created equal. "Now we are engaged\nin a great civil war,
testing whether that nation, or any nation so conceived and so dedicated, can\nlong endure.
We are met on a great battlefield of that war. We have come to dedicate a portion of\nthat field
as a final resting place for those who here gave their lives that that nation might live. It is
\naltogether fitting and proper that we should do this. "But in a larger sense we cannot dedicate,
we\ncannot consecrate, we cannot hallow this ground. The brave men, living and dead, who
struggled\nhere have consecrated it, far above our poor power to add or detract. The world
will little note, nor\nlong remember, what we say here, but it can never forget what they did
here. It is for us the living,\nrather, to be dedicated here to the unfinished work which they who
fought here have thus far so\nnobly advanced. It is rather for us to be here dedicated to the great
task remaining before us,that\nfrom these honored dead we take increased devotion to that cause
for which they gave the last full\nmeasure of devotion, that we here highly resolve that these dead
shall not have died in vain, that\nthis nation, under God, shall have a new birth of freedom, and
that government of the people, by the\npeople, for the people, shall not perish from the earth."
\n\n\n\n\nGettysburg address delivered at Gettysburg Pa. Nov. 19th, 1863. [n. p. n. d.].
http://www.loc.gov/resource/rbpe.24404500\n"
```

On The Web (Static)

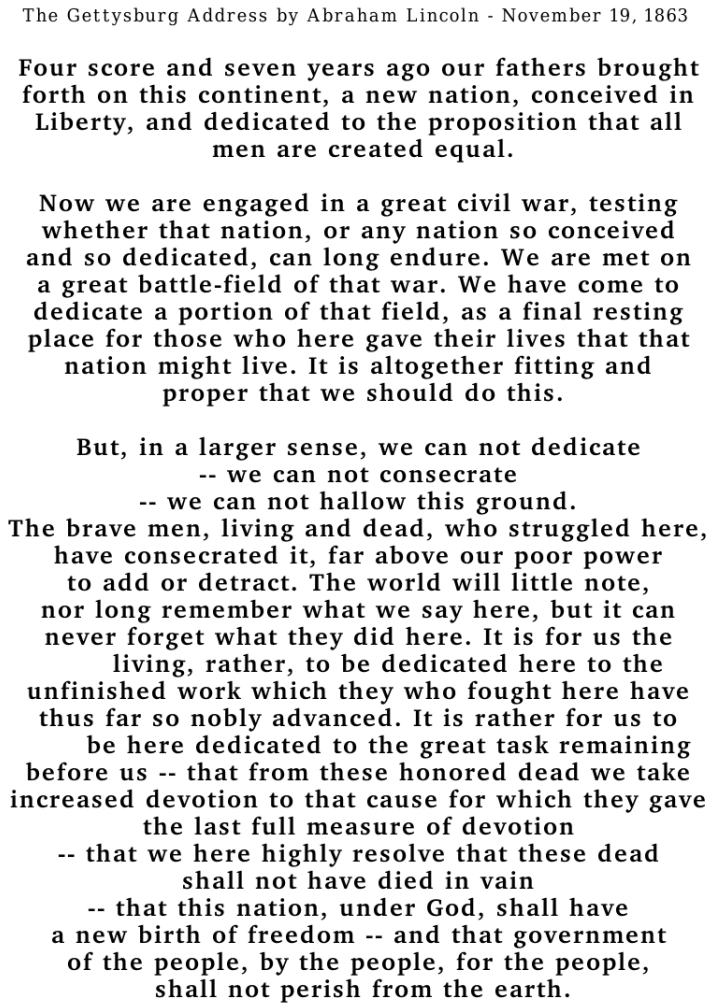
Read from a URL:

```
> GBA.3<-read_file("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/GBA.txt")
> GBA.3
.
.
.
> GBA.4<-pdf_text("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/GBA.pdf")
> GBA.4
[1] "Gettysburg address delivered at Gettysburg Pa. Nov. 19th, 1863. [n. p. n. d.].\nGettysburg
Address\n\nDelivered at Gettysburg, Pa.\n\nNov. 19th 1863.\n\n"Four score and seven years
ago our fathers brought forth on this continent a new nation, conceived\nin liberty, and dedicated
to the proposition that all men are created equal. "Now we are engaged\nin a great civil war,
testing whether that nation, or any nation so conceived and so dedicated, can\nlong endure.
We are met on a great battlefield of that war. We have come to dedicate a portion of\nthat field
as a final resting place for those who here gave their lives that that nation might live. It is
\naltogether fitting and proper that we should do this. "But in a larger sense we cannot dedicate,
we\ncannot consecrate, we cannot hallow this ground. The brave men, living and dead, who
struggled\nhere have consecrated it, far above our poor power to add or detract. The world
will little note, nor\nlong remember, what we say here, but it can never forget what they did
here. It is for us the living,\nrather, to be dedicated here to the unfinished work which they who
fought here have thus far so\nnobly advanced. It is rather for us to be here dedicated to the great
task remaining before us,that\nfrom these honored dead we take increased devotion to that cause
for which they gave the last full\nmeasure of devotion, that we here highly resolve that these dead
shall not have died in vain, that\nthis nation, under God, shall have a new birth of freedom, and
that government of the people, by the\npeople, for the people, shall not perish from the earth."
\n\n\n\n\nGettysburg address delivered at Gettysburg Pa. Nov. 19th, 1863. [n. p. n. d.].
http://www.loc.gov/resource/rbpe.24404500\n"
```

Handling Images: OCR

"Optical Character Recognition" translates images of text into machine-readable form.

Example image file (`GBA.png`):

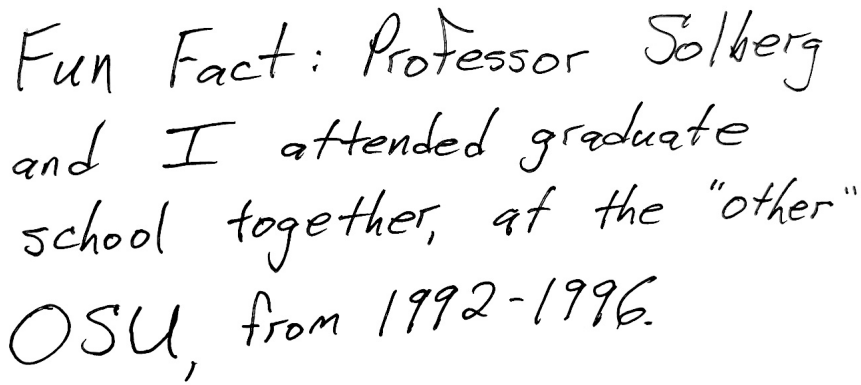


OCR (continued)

```
> eng<-tesseract("eng")
> GBA.5<-tesseract::ocr("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/GBA.png",
+                       engine = eng)

> cat(GBA.5)
The Gettysburg Address by Abraham Lincoln - November 19, 1863
Four score and seven years ago our fathers brought
forth on this continent, a new nation, conceived in
Liberty, and dedicated to the proposition that all
men are created equal.

Now we are engaged in a great civil war, testing
whether that nation, or any nation so conceived
and so dedicated, can long endure. We are met on
a great battle-field of that war. We have come to
dedicate a portion of that field, as a final resting
place for those who here gave their lives that that
nation might live. It is altogether fitting and
proper that we should do this.

But, in a larger sense, we can not dedicate
-- we can not consecrate
-- we can not hallow this ground.

The brave men, living and dead, who struggled here,
have consecrated it, far above our poor power
to add or detract. The world will little note,
nor long remember what we say here, but it can
never forget what they did here. It is for us the

living, rather, to be dedicated here to the
unfinished work which they who fought here have
thus far so nobly advanced. It is rather for us to
be here dedicated to the great task remaining
before us -- that from these honored dead we take
increased devotion to that cause for which they gave
the last full measure of devotion
-- that we here highly resolve that these dead
shall not have died in vain
-- that this nation, under God, shall have
a new birth of freedom -- and that government
of the people, by the people, for the people,
shall not perish from the earth.
```

OCR's Limits

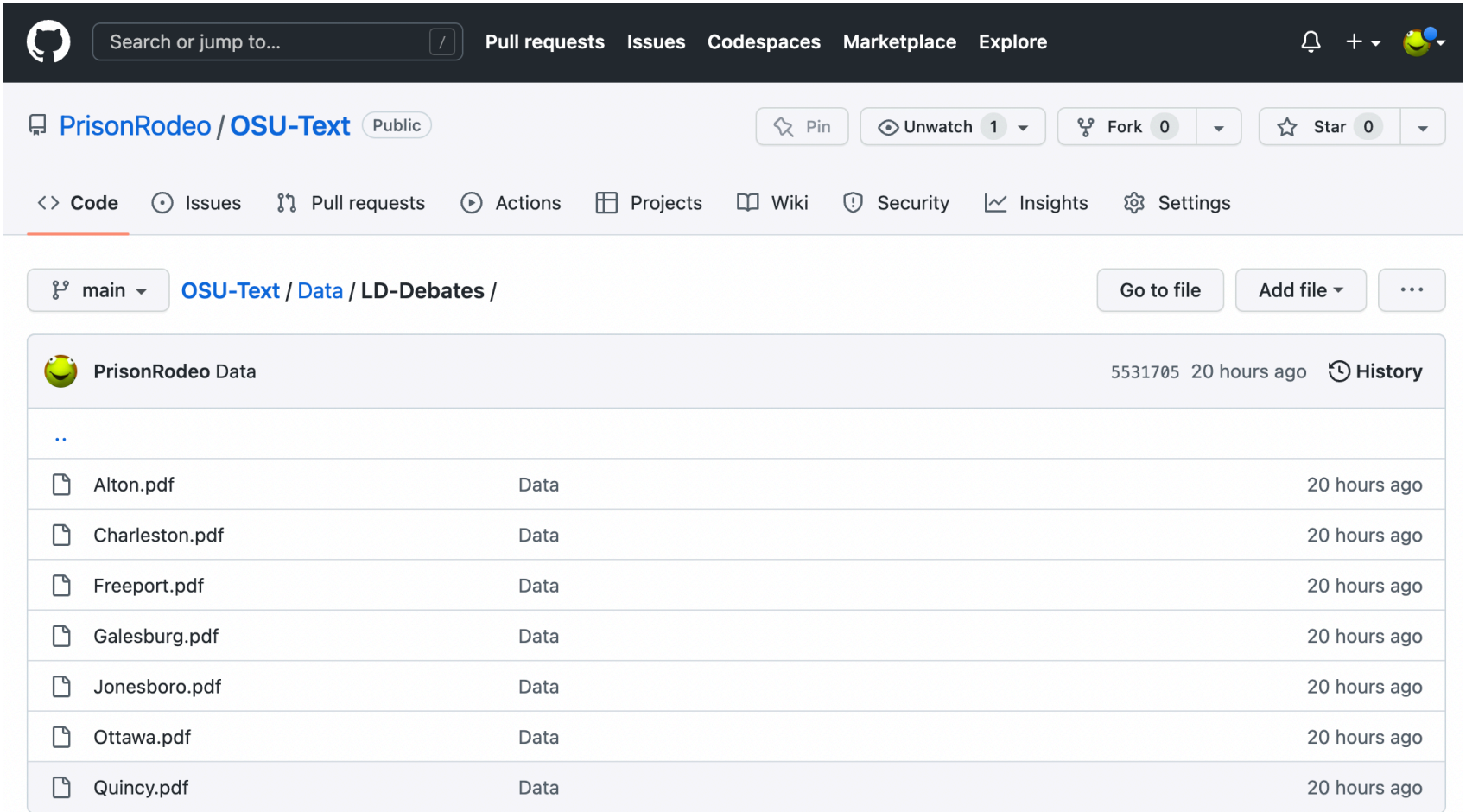
Handwriting:



OCR attempt:

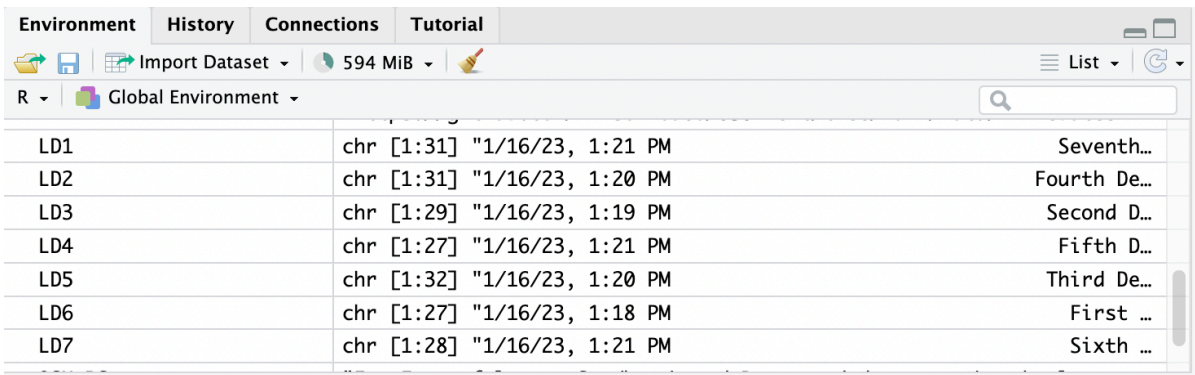
```
> OSU.RS<-tesseract::ocr("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/OSU-RS.jpg",
+                        engine = eng)

> cat(OSU.RS)
Fun Fact: folessor Se /bers
and Do attended greatate
school toge ther, at the "other"
OSU. trom (972-1796.
```

The Lincoln-Douglas Debates (1858):



Reading Many Files (continued)

```
> req<-GET("https://api.github.com/repos/PrisonRodeo/OSU-Text/contents/Data/LD-Debates")
> files<-httr::content(req)
> fnames<-sapply(files,function(x) x$name)
> fnames
[1] "Alton.pdf"      "Charleston.pdf" "Freeport.pdf"   "Galesburg.pdf"  "Jonesboro.pdf"
[6] "Ottawa.pdf"     "Quincy.pdf"

> for(i in i:length(fnames)){
+   fn<-paste0("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/LD-Debates/",
+              fnames[i]) # the URL of the specific file we want to get
+   f<-pdf_text(fn)       # get the file
+   out<-paste0("LD",i)   # create an object name for it
+   assign(out,f)         # give the object that name
+   rm(f)                 # housekeeping
+ }
```



A few things:

- To get tweets, you must sign up for a Twitter Developer account...
- Once you have that, you'll have to create a project, which will in turn give you a Twitter API Key and Secret
- These can then be used to scrape tweets (e.g., using the `rtweet` or `twitteR` packages)

Example:

```
library(twitteR)

consumer_key <- consumer_key_nt
consumer_secret <- consumer_secret_nt
access_token <- access_token_nt
access_secret <- access_secret_nt

setup_twitter_oauth(consumer_key, consumer_secret, access_token, access_secret)
```

Working With Text

Text as Data: Terminology

- *Word / Term*: In NLP, a single collection of letters signifying some meaning(s).
- *N-gram*: A collection of two or more words, treated as a unit / term.
- *Document*: A natural collection of terms with a common theme or content.
- *Tokenizing*: Breaking up a document into words, N-grams, sentences, or other syntactic subunits.
- *Corpus*: A collection of documents.
- *Stop Words*: A group of extremely common words typically of little direct interest to the researcher (e.g., conjunctions).
- *Normalization*: The creation of *equivalence classes* of terms. Examples include:
  - *Case folding*: Harmonizing the case/capitalization of terms (e.g., "Work" and "work")
  - *Stemming*: Reducing words with common stems to those stems (e.g., "works" and "working" become "work*")
  - *Lemmatization*: Similar to stemming: Combining words with common roots but more diverse meanings (e.g., "democracy" and "democratization").

Some Notation...

In general we structure text:

- N *unique* terms/words/tokens T_i in the corpus...
- ...indexed by $i = \{1, 2, ...N\}$
- J documents D_j, $j = \{1, 2, ...J\}$
- $X_{ij} =$ the ith unique term in the jth document in corpus $\mathbf{X}$

Capitalization and Punctuation

Capitalization / case-folding:

- Generally best removed (*Ferrari* and *ferrari* mean the same thing in English)
- Exceptions / potential pitfalls:
  - Proper nouns ("Mark Cuban" $\neq$ "mark" "cuban")
  - Acronyms ("CAT" $\neq$ "cat," etc.)
- Alternative: "truecasing"...

Punctuation:

- Periods, commas, colons, semicolons can usually go...
- Occasionally question marks and exclamation points are useful (e.g., sentiment analysis)
- **Order is important!** Don't remove punctuation prior to (say) tokenizing sentences...

Terms, Stems, and N-grams

Terms are the "lowest-level unit;" can be words, stems/roots, synonym groups, etc.

Stemming...

- Industry standard is the "snowball" stemmer...
- Details at `http://snowballstem.org/`

N-grams:

- Can be specified/user-defined ("Utah Jazz," "Orlando Magic," etc.)
- Useful for proper nouns, terms of art, etc.
- Can also be built from the corpus ("shingled")

Stop Words

Stop words are common words that rarely have intrinsic meaning...

- We *usually* want to remove them...
- Standard R stop words:

```
> stopwords("en")
  [1] "a"     "an"    "and"   "are"   "as"
  [6] "at"    "be"    "but"   "by"    "for"
 [11] "if"    "in"    "into"  "is"    "it"
 [16] "no"    "not"   "of"    "on"    "or"
 [21] "such"  "that"  "the"   "their" "then"
 [26] "there" "these" "they"  "this"  "to"
 [31] "was"   "will"  "with"
```

- Other lists are much longer, e.g.
  https://github.com/stopwords-iso/stopwords-iso/
- Potential issues:
  - Proper nouns ("The Who," "That Was Then")
  - Stop word lists often have gendered pronouns (Monroe, Colaresi, and Quinn 2008)
  - Any word *can* be a stop word...

A Toy Example

A single line from the Gettysburg Address:

```
> GBAC<-"It is rather for us to be here dedicated to the
+        great task remaining before us -- that from these
+        honored dead we take increased devotion to that
+        cause for which they gave the last full measure of
+        devotion -- that we here highly resolve that these
+        dead shall not have died in vain -- that this nation,
+        under God, shall have a new birth of freedom --
+        and that government of the people, by the people,
+        for the people, shall not perish from the earth."


> GBAC

[1] "It is rather for us to be here dedicated to the \n       great task remaining before us --
that from these \n       honored dead we take increased devotion to that \n       cause for which
they gave the last full measure of \n       devotion -- that we here highly resolve that these \n
       dead shall not have died in vain -- that this nation, \n       under God, shall have a new
birth of freedom -- \n       and that government of the people, by the people, \n       for the
people, shall not perish from the earth."
```

Some Transformations

Make lower-case:

```
> tolower(GBAC)

[1] "it is rather for us to be here dedicated to the \n       great task remaining before us --
that from these \n       honored dead we take increased devotion to that \n       cause for which
they gave the last full measure of \n       devotion -- that we here highly resolve that these \n
dead shall not have died in vain -- that this nation, \n       under god, shall have a new
birth of freedom -- \n       and that government of the people, by the people, \n       for the
people, shall not perish from the earth."
```

Make upper-case:

```
> toupper(GBAC)

[1] "IT IS RATHER FOR US TO BE HERE DEDICATED TO THE \n       GREAT TASK REMAINING BEFORE US --
THAT FROM THESE \n       HONORED DEAD WE TAKE INCREASED DEVOTION TO THAT \n       CAUSE FOR
WHICH THEY GAVE THE LAST FULL MEASURE OF \n       DEVOTION -- THAT WE HERE HIGHLY RESOLVE
THAT THESE \n       DEAD SHALL NOT HAVE DIED IN VAIN -- THAT THIS NATION, \n       UNDER GOD, SHALL
HAVE A NEW BIRTH OF FREEDOM -- \n       AND THAT GOVERNMENT OF THE PEOPLE, BY THE PEOPLE,
\n       FOR THE PEOPLE, SHALL NOT PERISH FROM THE EARTH."
```

Make “title-case”:

```
> str_to_title(GBAC)

[1] "It Is Rather For Us To Be Here Dedicated To The \n       Great Task Remaining Before Us --
That From These \n       Honored Dead We Take Increased Devotion To That \n       Cause For
Which They Gave The Last Full Measure Of \n       Devotion -- That We Here Highly Resolve
That These \n       Dead Shall Not Have Died In Vain -- That This Nation, \n       Under God, Shall
Have A New Birth Of Freedom -- \n       And That Government Of The People, By The People,
\n       For The People, Shall Not Perish From The Earth."
```

More Transformations

Replace characters:

```
> chartr("a","A",GBAC)

[1] "It is rAther for us to be here dedicAted to the \n       greAt tAsk remAining before us --
thAt from these \n       honored deAd we tAke increAsed devotion to thAt \n       cAuse for
which they gAve the lAst full meAsure of \n       devotion -- thAt we here highly resolve
thAt these \n       deAd shAll not hAve died in vAin -- thAt this nAtion, \n       under God, shAll
hAve A new birth of freedom -- \n       And thAt government of the people, by the people,
\n       for the people, shAll not perish from the eArth."
```

Remove punctuation:

```
> removePunctuation(GBAC)

[1] "It is rather for us to be here dedicated to the \n       great task remaining before us
that from these \n       honored dead we take increased devotion to that \n       cause for
which they gave the last full measure of \n       devotion  that we here highly resolve
that these \n       dead shall not have died in vain  that this nation \n       under God shall
have a new birth of freedom  \n       and that government of the people by the people
\n       for the people shall not perish from the earth"
```

Remove specific words:

```
> removeWords(GBAC, c("us","that"))

[1] "It is rather for  to be here dedicated to the \n       great task remaining before  --
from these \n       honored dead we take increased devotion to  \n       cause for which
they gave the last full measure of \n       devotion --  we here highly resolve  these \n
dead shall not have died in vain --  this nation, \n       under God, shall have a new birth
of freedom -- \n       and  government of the people, by the people, \n       for the people,
shall not perish from the earth."
```

"Tokenizing" Sentences

Retain sentence structure:

```
> GBA<-stripWhitespace(GBA)
>
> GBA.sent<-tokenize_sentences(GBA)
>
> GBA.sent
[[1]]
 [1] "Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty,
 [2] "Now we are engaged in a great civil war, testing whether that nation, or any nation so conceived and dedicated,
 [3] "We are met on a great battle-field of that war."
 [4] "We have come to dedicate a portion of that field, as a final resting place for those who here gave their lives
 [5] "It is altogether fitting and proper that we should do this."
 [6] "But, in a larger sense, we can not dedicate -- we can not consecrate -- we can not hallow -- this ground."
 [7] "The brave men, living and dead, who struggled here, have consecrated it, far above our poor power to add or det
 [8] "The world will little note, nor long remember what we say here, but it can never forget what they did here."
 [9] "It is for us the living, rather, to be dedicated here to the unfinished work which they who fought here have th
[10] "It is rather for us to be here dedicated to the great task remaining before us -- that from these honored dead
```

How many sentences are there?

```
> length(GBA.sent[[1]])
[1] 10
```

More Tokenizing

Tokenize words:

```
> GBA.words <- tokenize_words(GBA)
> GBA.words
[[1]]
  [1] "four"        "score"       "and"         "seven"       "years"       "ago"         "our"         "fathers"     "brought"     "forth"
 [11] "on"          "this"        "continent"   "a"           "new"         "nation"      "conceived"   "in"          "liberty"     "and"
.
.
.
[261] "the"         "people"      "for"         "the"         "people"      "shall"       "not"         "perish"      "from"        "the"
[271] "earth"

> length(GBA.words[[1]]) # total word count
[1] 271
```

Turn sentences into words (nested list):

```
> GBA.sw <- tokenize_words(GBA.sent[[1]])
> GBA.sw
[[1]]
 [1] "four"        "score"       "and"         "seven"       "years"       "ago"         "our"         "fathers"     "brought"
[10] "forth"       "on"          "this"        "continent"   "a"           "new"         "nation"      "conceived"   "in"
.
.
.
[[10]]
 [1] "it"          "is"          "rather"      "for"         "us"          "to"          "be"          "here"        "dedicated"
[10] "to"          "the"         "great"       "task"        "remaining"   "before"      "us"          "that"        "from"
[19] "these"       "honored"     "dead"        "we"          "take"        "increased"   "devotion"    "to"          "that"
[28] "cause"       "for"         "which"       "they"        "gave"        "the"         "last"        "full"        "measure"
[37] "of"          "devotion"    "that"        "we"          "here"        "highly"      "resolve"     "that"        "these"
[46] "dead"        "shall"       "not"         "have"        "died"        "in"          "vain"        "that"        "this"
[55] "nation"      "under"       "god"         "shall"       "have"        "a"           "new"         "birth"       "of"
[64] "freedom"     "and"         "that"        "government"  "of"          "the"         "people"      "by"          "the"
[73] "people"      "for"         "the"         "people"      "shall"       "not"         "perish"      "from"        "the"
[82] "earth"
>
> # Count words per sentence:
>
> sapply(GBA.sw, length)
 [1] 30 23 11 27 11 19 21 21 26 82
```

Stop-Word Removal and Stemming

Remove "stop words":

```
> removeWords(GBA,stopwords("en"))

[1] "Four score  seven years ago  fathers brought forth  continent,  new nation, conceived  Liberty,  dedicated
proposition  men  created equal. Now  engaged  great civil war, testing whether  nation,  nation  conceived
dedicated, can long endure. We  met  great battle-field  war. We  come  dedicate  portion  field,  final
resting place  gave  lives  nation might live. It  altogether fitting  proper  . But,  larger sense,  can
dedicate --  can  consecrate --  can  hallow --  ground. The brave men, living  dead,  struggled ,  consecrated ,
far  poor power  add  detract. The world  little note,  long remember  say ,  can never forget  . It  us
living, rather,  dedicated  unfinished work  fought  thus far  nobly advanced. It  rather  us  dedicated
great task remaining  us --  honored dead  take increased devotion  cause  gave  last full measure
devotion --  highly resolve  dead shall  died  vain --  nation,  God, shall  new birth  freedom --
government  people,  people,  people, shall  perish  earth. "
```

"Stem" the words:

```
> stemDocument(GBA)

[1] "Four score and seven year ago our father brought forth on this continent, a new nation, conceiv in
Liberty, and dedic to the proposit that all men are creat equal. Now we are engag in a great civil war,
test whether that nation, or ani nation so conceiv and dedicated, can long endure. We are met on a great
battle-field of that war. We have come to dedic a portion of that field, as a final rest place for those who
here gave their live that that nation might live. It is altogeth fit and proper that we should do this. But, in
a larger sense, we can not dedic -- we can not consecr -- we can not hallow -- this ground. The brave
men, live and dead, who struggl here, have consecr it, far abov our poor power to add or detract. The
world will littl note, nor long rememb what we say here, but it can never forget what they did here. It is
for us the living, rather, to be dedic here to the unfinish work which they who fought here have thus far
so nobli advanced. It is rather for us to be here dedic to the great task remain befor us -- that from these
honor dead we take increas devot to that caus for which they gave the last full measur of devot -- that
we here high resolv that these dead shall not have die in vain -- that this nation, under God, shall have
a new birth of freedom -- and that govern of the people, by the people, for the people, shall not perish
from the earth."
```

Count Some Words

Term-document matrices (and document-term matrices) are for word counts...

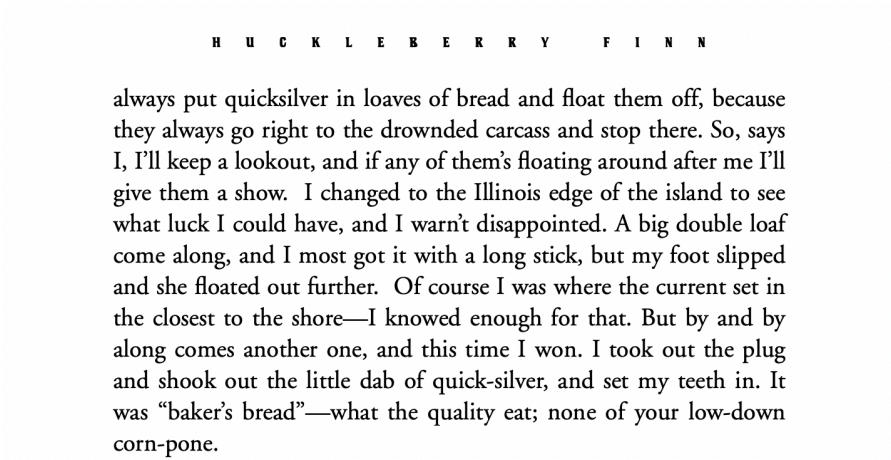
- A *term-document matrix* has:
  - N rows, corresponding to the N unique terms in the corpus
  - J columns, corresponding to the J documents in the corpus
  - Entries N_{ij} that represent the number of times term i appears in document j

- A *document-term matrix* is a transposed term-document matrix

The Adventures of Huckleberry Finn

From https://contentserver.adobe.com/store/books/HuckFinn.pdf...

Note the page headings:



H U C K L E B E R R Y F I N N

always put quicksilver in loaves of bread and float them off, because they always go right to the drownded carcass and stop there. So, says I, I'll keep a lookout, and if any of them's floating around after me I'll give them a show. I changed to the Illinois edge of the island to see what luck I could have, and I warn't disappointed. A big double loaf come along, and I most got it with a long stick, but my foot slipped and she floated out further. Of course I was where the current set in the closest to the shore—I knowed enough for that. But by and by along comes another one, and this time I won. I took out the plug and shook out the little dab of quick-silver, and set my teeth in. It was "baker's bread"—what the quality eat; none of your low-down corn-pone.

which give us:

```
...what a mess you are always making!" The widow put in a good word for me, but that warn't
going to keep off the bad luck, I knowed that well enough. I started out, after breakfast, feeling
worried and shaky, and wondering where it was going to fall on me, and what it was 16 "
[24] " H U C K L E B E R R Y F I N N "There; you see it says `for a consideration.' That means
I have bought it of you and paid you for it. Heres a dollar for you. Now you sign it." So I signed it,...
```

More Adventures with Huckleberry Finn

Text preprocessing:

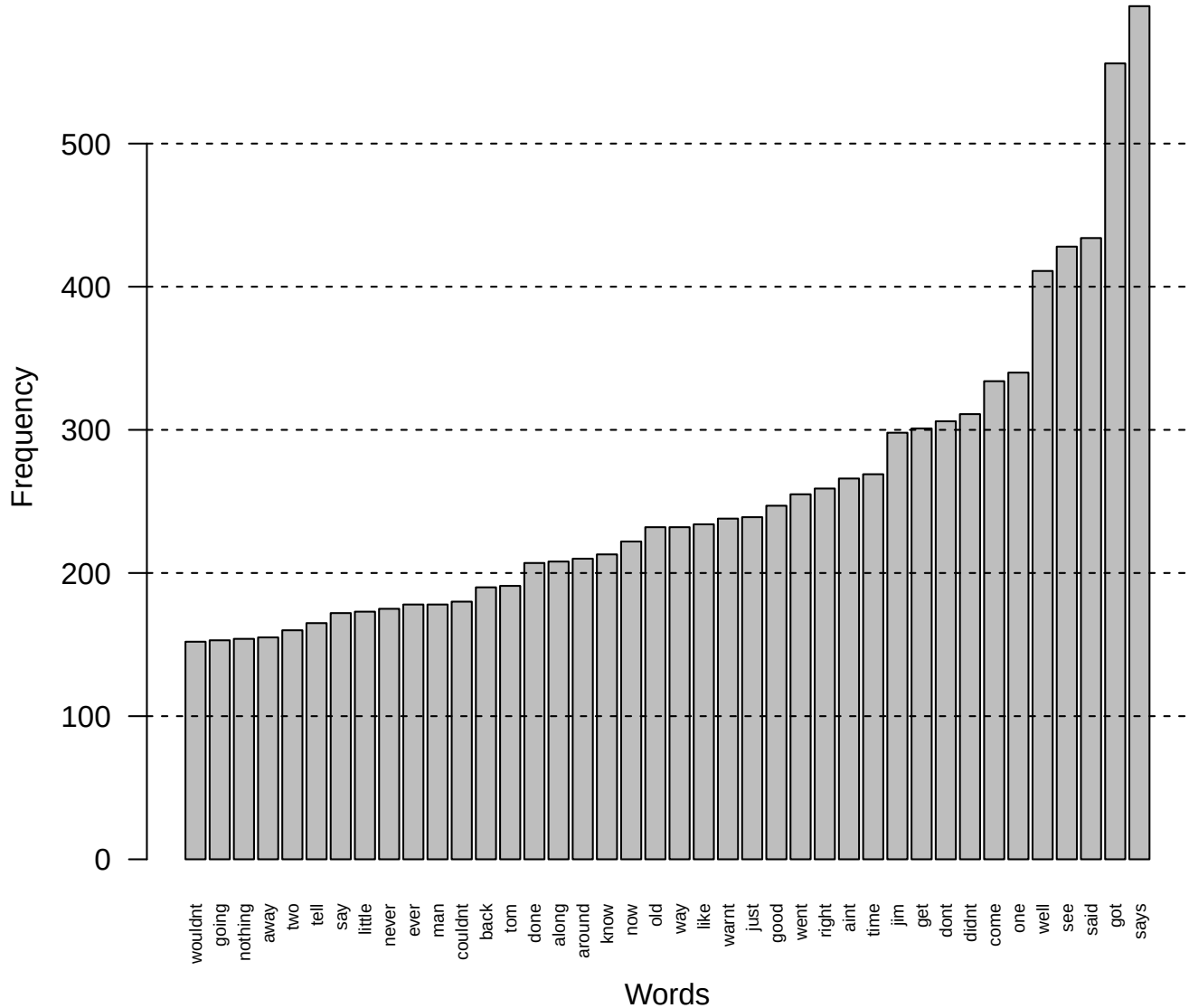
```
> # Clean up that document...
> #
> # First: remove everything that isn't an
> # alphanumeric symbol:
>
> removeSpecialChars <- function(x) gsub("[^a-zA-Z0-9 ]","",x)
> AHF<-removeSpecialChars(AHF)
>
> # Now text processing:
>
> AHF.C<-VCorpus(VectorSource(AHF)) # create a "corpus"
> AHF.C<-tm_map(AHF.C,content_transformer(tolower))
> AHF.C<-tm_map(AHF.C,content_transformer(removeNumbers))
> AHF.C<-tm_map(AHF.C,content_transformer(removePunctuation))
> AHF.C<-tm_map(AHF.C,removeWords,stopwords('en'))
> AHF.C<-tm_map(AHF.C,stripWhitespace)
> AHF.C<-tm_map(AHF.C,removeWords,"H U C K L E B E R R Y F I N N") # strip page headings
```

AHF: Word Frequencies

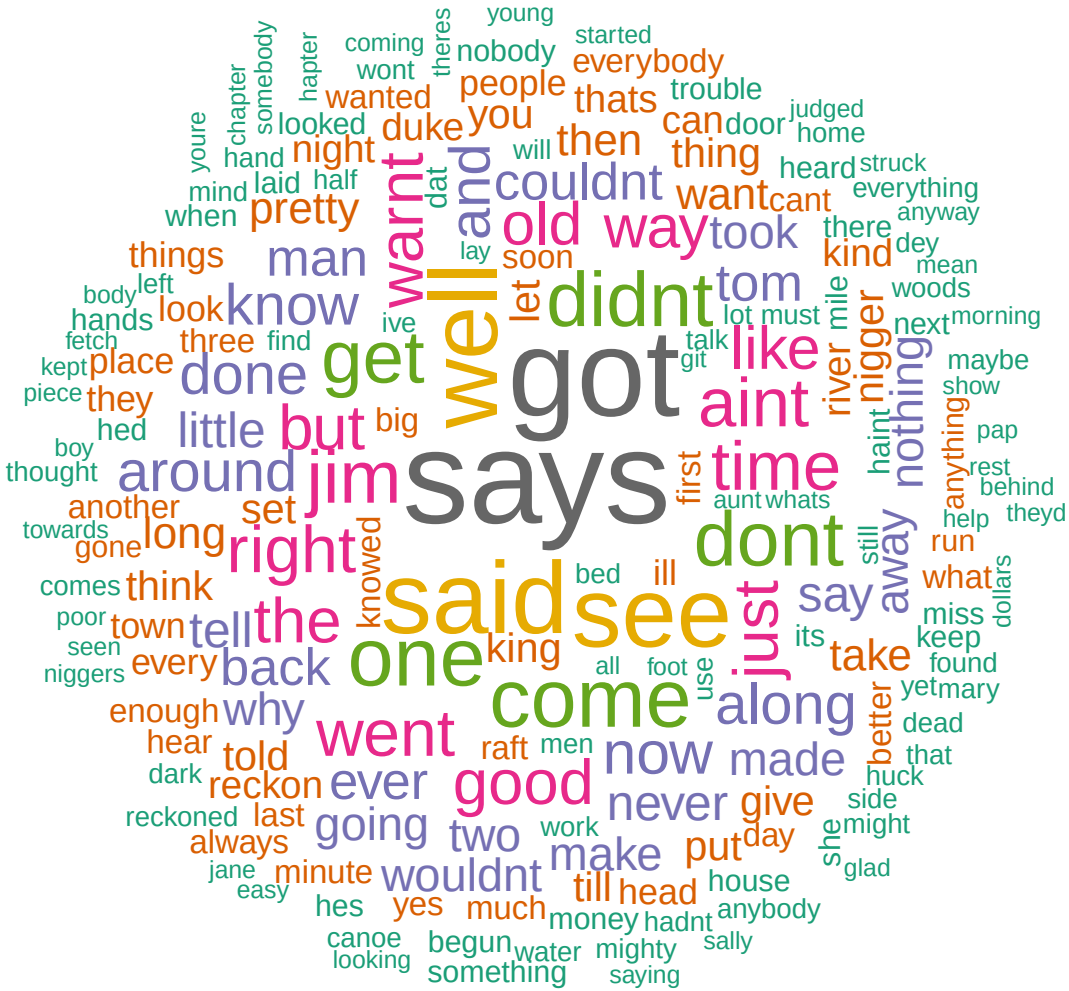
Words that appear ≥ 100 times:

```
> AHF.TDM<-TermDocumentMatrix(AHF.C)

> findFreqTerms(AHF.TDM,100,Inf)
 [1] "aint"    "along"   "around"  "away"    "back"    "can"     "come"    "couldnt"
 [9] "didnt"   "done"    "dont"    "duke"    "ever"    "get"     "give"    "going"
[17] "good"    "got"     "ill"     "jim"     "just"    "kind"    "king"    "know"
[25] "let"     "like"    "little"  "long"    "look"    "made"    "make"    "man"
[33] "never"   "nigger"  "night"   "nothing" "now"     "old"     "one"     "people"
[41] "pretty"  "put"     "reckon"  "right"   "river"   "said"    "say"     "says"
[49] "see"     "set"     "take"    "tell"    "thats"   "thing"   "things"  "think"
[57] "till"    "time"    "told"    "tom"     "took"    "two"     "want"    "warnt"
[65] "way"     "well"    "went"    "wouldnt"
```

AHF: Top 40 Word Frequencies



AHF: Word Cloud!



For more on word clouds in R, see (e.g.) here.

Term "Importance" (TF v. TF-IDF)

Term frequency:

$$N_{ij} = \text{The number of times term } i \text{ appears in document } j$$

Term frequency (normalized for document length):

$$TF_{ij} = \frac{N_{ij}}{\sum_{i=1}^{N} N_{ij}},$$

is the fraction of all terms in D_j that are term T_i.

Inverse document frequency (normalized) is:

$$IDF_i = \log_2 \frac{J}{J_i}$$

where J_i is the number of documents in which T_i occurs.

TF-IDF$_{ij}$ is then simply $TF_{ij} \times IDF_i$

TF-IDF Toy Example

Three "documents":

$$
\begin{aligned}
A &= \{\text{red, blue, red}\} \\
B &= \{\text{green, blue, orange}\} \\
C &= \{\text{yellow, blue, yellow}\}
\end{aligned}
$$

Example one:

- In document A "red" appears twice ($TF_{ij} = 2$), and
- "red" is two of the three total terms in that document (normed $TF_{ij} = 0.67$)
- "red" appears in only one of the three documents ($IDF_i = log_2[3/1] = 1.6$)
- The TF-IDF for "red" in document A is $0.67 \times 1.6 \approx 1.06$

Example two:

- In document C "blue" appears once ($TF_{ij} = 1$), and
- "blue" is one of the three total terms in that document (normed $TF_{ij} = 0.33$)
- "blue" appears in all three documents ($IDF_i = log_2[3/3] = 0$)
- The TF-IDF for "blue" in document C is $0.33 \times 0 = 0$

TF-IDF Intuition

In general:

- (Normalized) TF indicates the prevalence of a term in a document
- IDF reflects how common or rare the word is across documents
- IDF is thus a measure of the level of "informativeness" (or "document-specificity") of a word
- TF-IDF is thus a measure of a term's **"importance"** (in some respects)
- Note as well that, while useful, TF-IDF has some issues to pay attention to

TF-IDF Matrix Toy Example

From the example above:

```
> A<-"red blue red"
> B<-"green blue orange"
> C<-"yellow blue yellow"
>
> TDM<-TermDocumentMatrix(c(A,B,C))
> TDM2<-as.matrix(TDM)
> TDM2
        Docs
Terms    1 2 3
  blue   1 1 1
  red    2 0 0
  green  0 1 0
  orange 0 1 0
  yellow 0 0 2
```

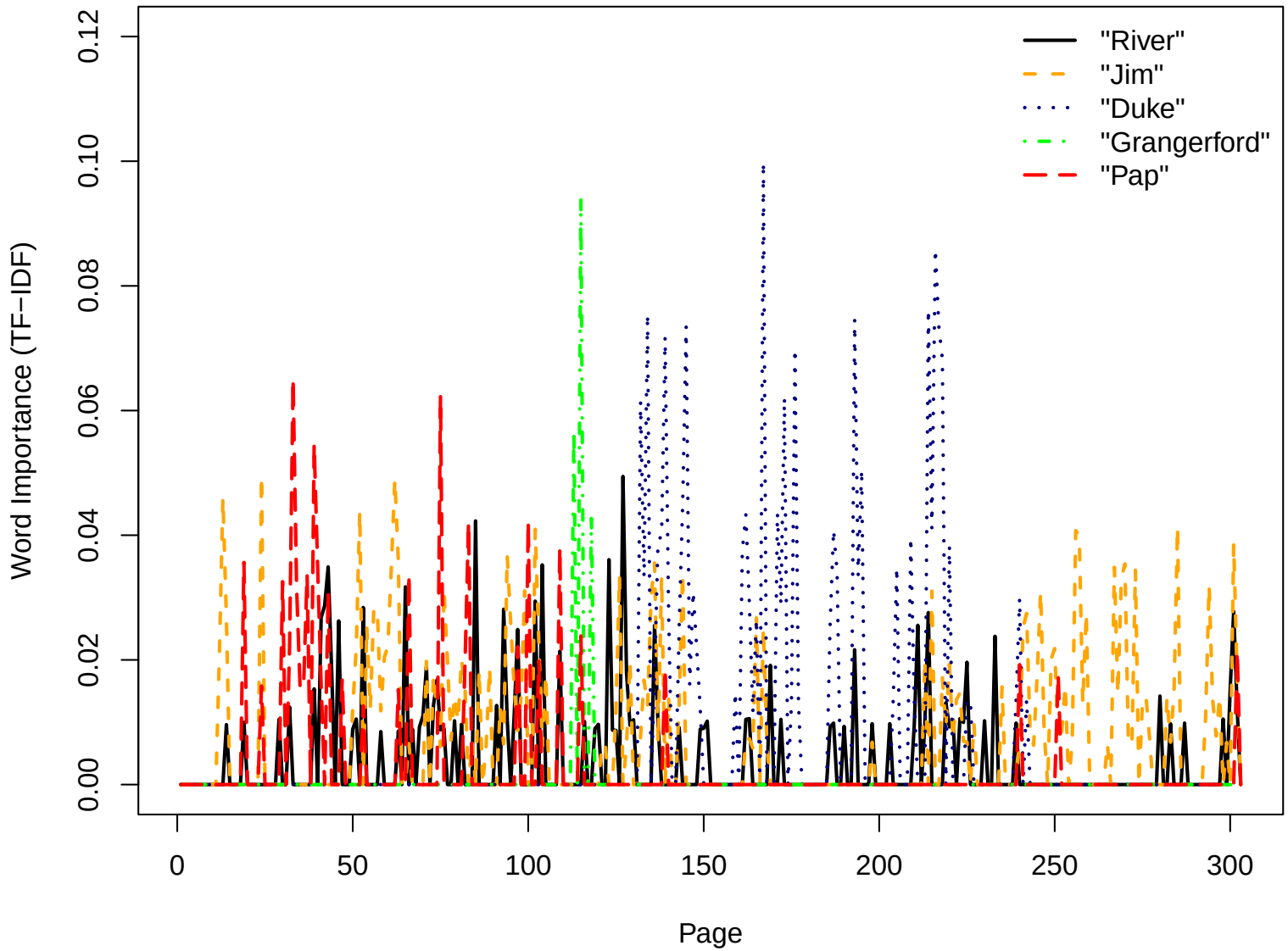
The TF-IDF matrix:

```
> TFIDF<-as.matrix(weightTfIdf(TDM))
> TFIDF
        Docs
Terms            1         2        3
  blue   0.000000 0.0000000 0.000000
  red    1.056642 0.0000000 0.000000
  green  0.000000 0.5283208 0.000000
  orange 0.000000 0.5283208 0.000000
  yellow 0.000000 0.0000000 1.056642
```

Word Importance: TF-IDF

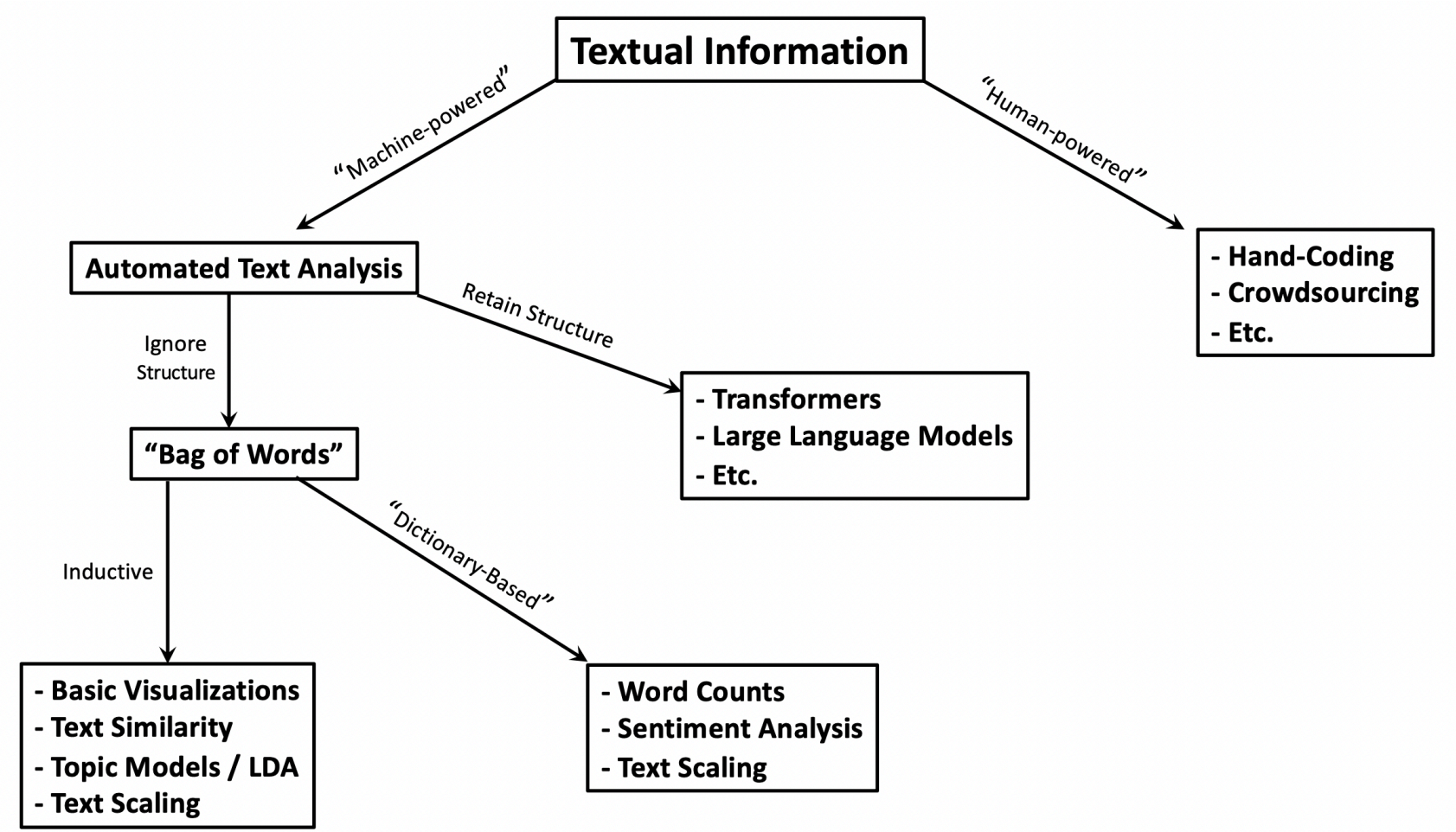
```
> AHF.TFIDF<-weightTfIdf(AHF.TDM) # create TF-IDF matrix
> str(AHF.TFIDF)

List of 6
 $ i       : int [1:40747] 139 3307 5871 6827 10564 3307 4606 139 2118 3307 ...
 $ j       : int [1:40747] 1 1 1 1 1 2 2 3 3 3 ...
 $ v       : Named num [1:40747] 1.087 0.831 1.049 1.649 1.449 ...
  ..- attr(*, "names")= chr [1:40747] "1" "1" "1" "1" ...
 $ nrow    : int 11698
 $ ncol    : int 303
 $ dimnames:List of 2
  ..$ Terms: chr [1:11698] "aaamen" "ababy" "abank" "abar" ...
  ..$ Docs : chr [1:303] "1" "2" "3" "4" ...
 - attr(*, "class")= chr [1:2] "TermDocumentMatrix" "simple_triplet_matrix"
 - attr(*, "weighting")= chr [1:2] "term frequency - inverse document frequency (normalized)" "tf-
idf"

> M<-as.matrix(AHF.TFIDF) # keep in a matrix...
> M2 <- data.frame(M)     # ...or a data frame
```

Word Importance: Plots, etc.



Bags of Words

A Typology


Textual Information
"Machine-powered"
"Human-powered"
Automated Text Analysis
- Hand-Coding
- Crowdsourcing
- Etc.
Retain Structure
Ignore Structure
- Transformers
- Large Language Models
- Etc.
"Bag of Words"
"Dictionary-Based"
Inductive
- Basic Visualizations
- Text Similarity
- Topic Models / LDA
- Text Scaling
- Word Counts
- Sentiment Analysis
- Text Scaling

Overview: Dictionary-Based Methods

- **Classification** task:
  - *Categorize* documents into classes C, and/or
  - *Score* documents degree of association with those classes.

- Heuristic: **Dictionaries assign weights to words / terms.**

- Formally: For $j \in \{1...J\}$ words in a corpus of $i = \{1...N\}$ documents, the *document score* is:

$$S_i = \frac{\sum_{j=1}^{J} \omega_j X_{ij}}{\sum_{j=1}^{J} X_{ij}}$$

  where
  - X_{ij} is the number of instances of word j in document i, and
  - ω_j is the weight assigned to each word by the dictionary.

General Dictionary-Based Methods: How-To

1. Obtain / preprocess documents (stemming, stop words, etc.)
2. Obtain / create a dictionary
3. Score documents by calculating S_i
   - Weights ω_j can be positive or negative
   - Words in the corpus but not in the dictionary have $\omega_j = 0$
4. (Optional:) Classify documents by mapping $S_i \rightsquigarrow C_i$

Toy Example: "Truthiness"

- Document: `{TRUE FALSE TRUE FALSE TRUE}`
- Dictionary:

Term	ω_j
`TRUE`	1.0
`FALSE`	0.0

- Word counts:

$$\mathbf{X} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

- Score:

$$S_i = \frac{(1.0 \times 3) + (0.0 \times 2)}{(3 + 2)} = \frac{3}{5} = 0.6$$

Dictionary-Based Classification Tasks

Some examples:

- Sentiment Analysis
  - What is the *emotional valence* of the documents?
  - What are the *emotions* expressed? (pity, anger, jealousy, etc.)
- Topic(s) / Topic Modeling
  - What are documents *about*?
  - What thing(s) are *emphasized*?
- Tone / Style
  - Authorship / provenance
  - Specialization of language (e.g., "hold harmless")

Sentiment Analysis

"...[C]omputational study of how opinions, attitudes, emotions, and perspectives are expressed in language..."
– Liu (2011)

A good overview is:

Pang, Bo, and Lillian Lee. 2008. "Opinion Mining and Sentiment Analysis." *Foundations and Trends in Information Retrieval* 2:1-135.

For example:

"Arizona bears the brunt of the country's illegal immigration problem. Its citizens feel themselves under siege by large numbers of illegal immigrants who invade their property, strain their social services, and even place their lives in jeopardy. Federal officials have been unable to remedy the problem, and indeed have recently shown that they are unwilling to do so."

– Justice Scalia, dissenting in *Arizona v. United States* (2012)

Where Do (Sentiment) Dictionaries Come From?

Sources for sentiment:

- "Standard" dictionaries
  - Code sentiment in common (contemporary, usually American) English
  - See below; there's a list here
- "By hand"...
  - Requires careful thought / luck / divine help
  - **Validate**. Seriously.
- "Crowdsourced" methods: RAs, MTurk, etc.
  - "On a scale from -10 to 10, how positive is the word...?"
  - Can be made context-specific, etc.
- Statistical approaches
  - Fit a model to some document-level outcome $\rightarrow$ most predictive words = dictionary
  - "Model" = lasso / ridge regression / elastic net, etc.
  - Again, **validation** is key...

Common (English) Sentiment Dictionaries

- General Inquirer (http://www.wjh.harvard.edu/~inquirer/)
- AFINN (http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=6010)
- QDAP dictionaries (https://cran.r-project.org/web/packages/qdap/index.html)
- WordStat (find it here)
- LIWC (http://liwc.wpengine.com/)

Sentiment Dictionary Examples

General Inquirer:

- Words scored either positive (+1) or negative (-1)
- 1637 positive words, 2005 negative words

AFINN (2477 total words, scored [-5,5]):

Term	ω_j
bastard	-5
bitch	-5
⋮	⋮
worn	-1
some kind	0
aboard	1
⋮	⋮
superb	5
thrilled	5

Sentiment Analysis Options in R

- `SentimentAnalysis`
  - Built by finance people → dictionaries, etc.
  - Plays well with `tm`
  - My current favorite (see the vignette)
- `tidyverse`, etc.
  - Requires admission to the cult of Wickham
  - Details here: `https://www.tidytextmining.com/`
  - Tons of tutorials (here, here, here, etc.)
- `RSentiment` (super minimal)
- `sentiment` (deprecated)

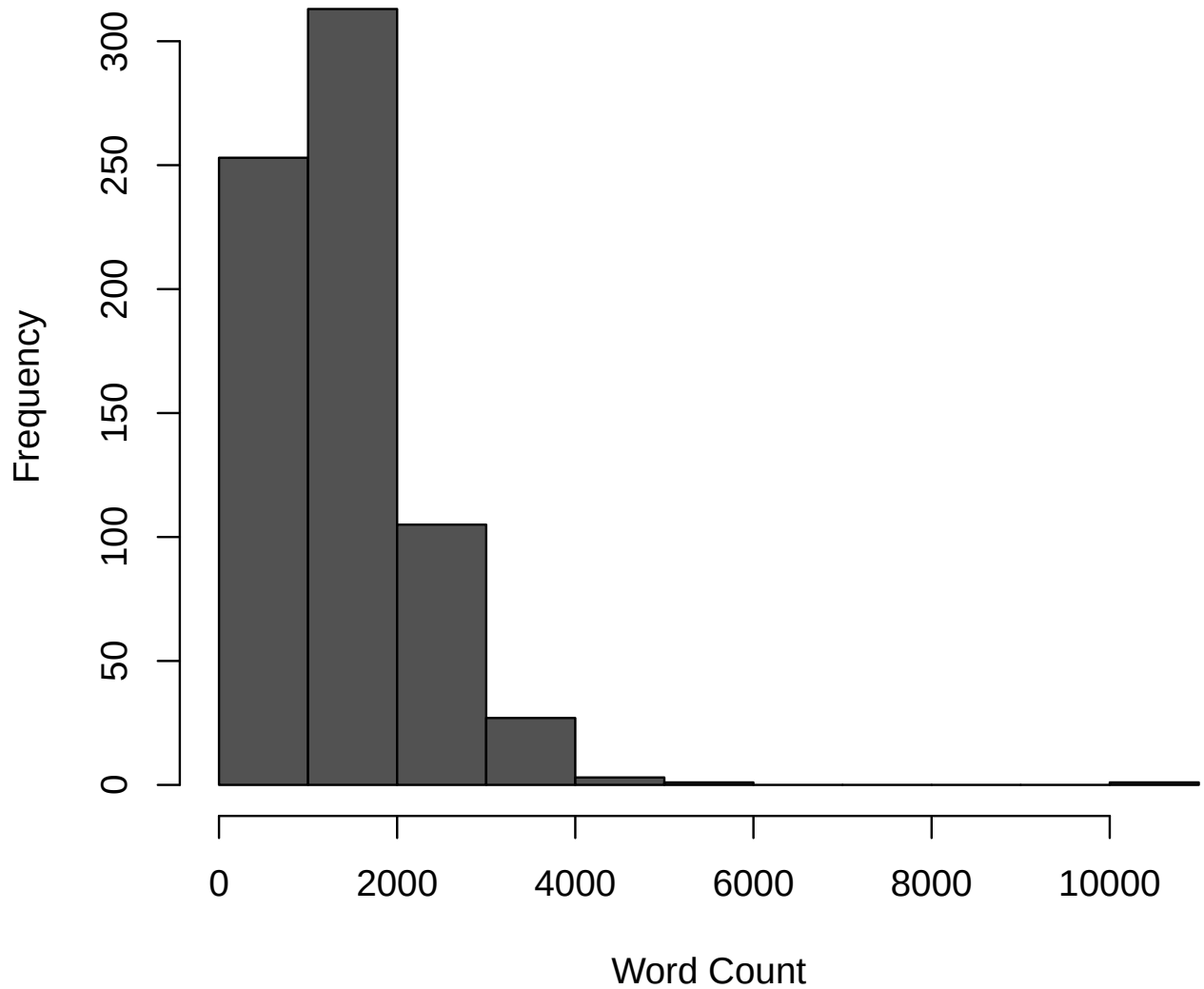
`SentimentAnalysis` Details

- Works with character objects, data frames, corpuses / TDMs / DTMs from `tm`
- Built-in dictionaries: General Inquirer, QDAP, two finance-specific (Henry 2008; Loughran and McDonald 2011)
- Can also create dictionaries "by hand" or through predictive power of words vis-a-vis some response (via glm, lasso, etc.)
- `analyzeSentiment` is the workhorse
  - Defaults to using all four built-in dictionaries
  - Stems and removes stop words by default
  - Outputs a `data.frame` with document-level sentiment scores
- Other useful things:
  - Built-in tokenizer / N-gram creator
  - Convert continuous sentiment scores to binary (0/1) or directional (-1/0/1) values
  - Can generate predictions and assess predictive performance...

Example: UNHCR Speeches

- All speeches made by the High Commissioner of the U.N. Refugee Agency, 1970-2016 ($N = 703$)
- Metadata include ID, speaker, title, and date
- Source: https://www.kaggle.com/franciscadias/un-refugee-speech-analysis/

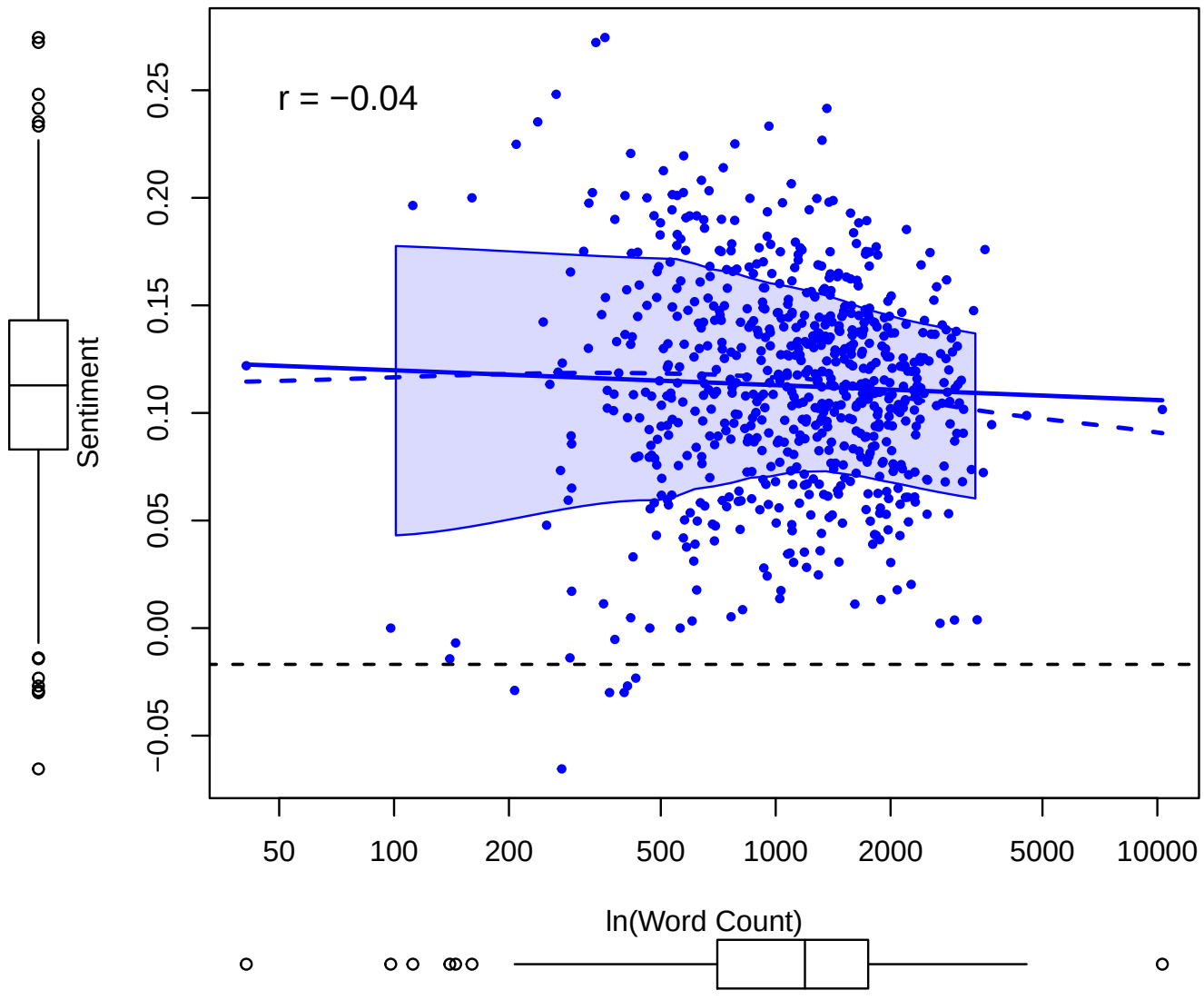
UNHCR Speech Word Counts, 1970-2016



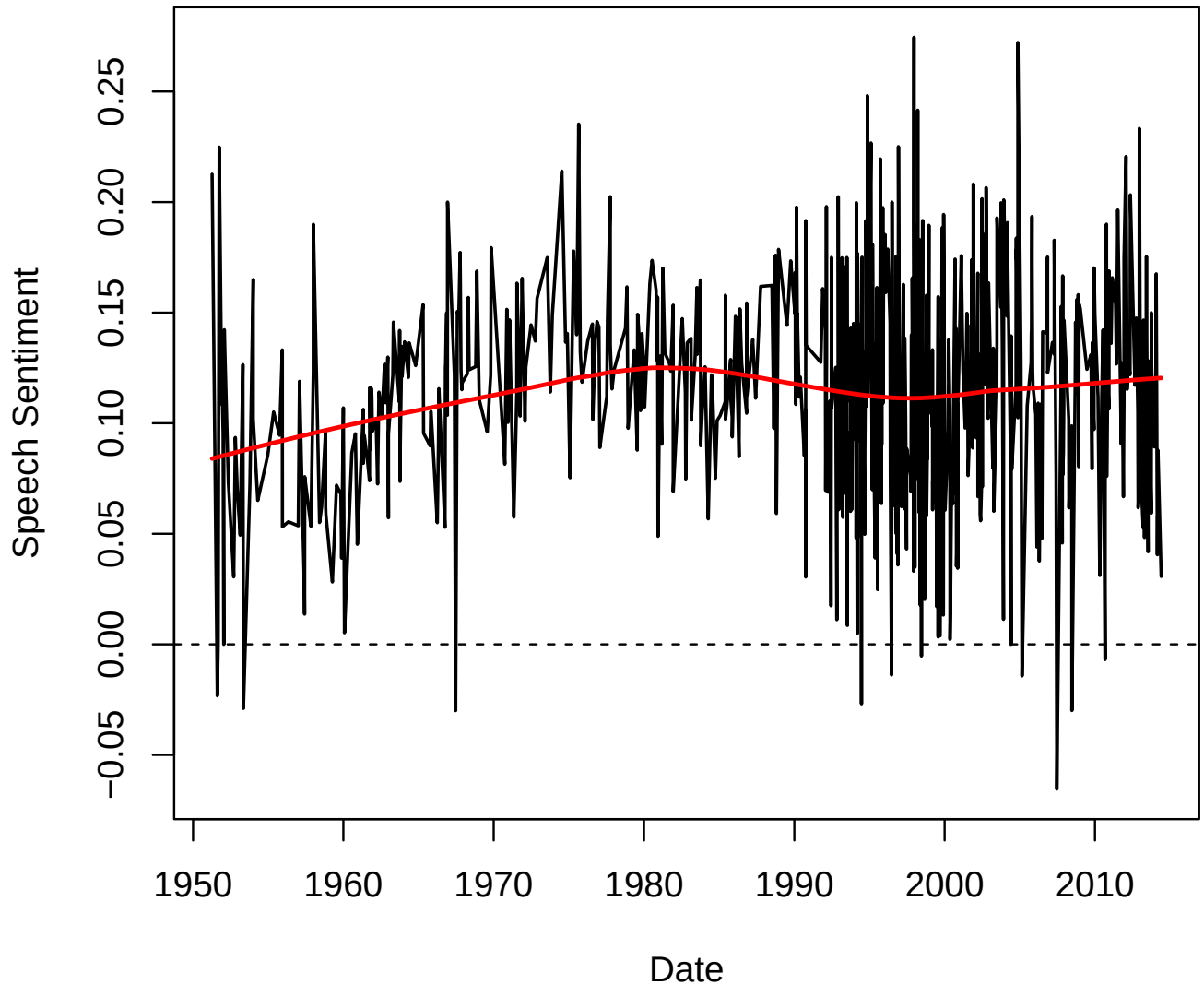
Simple Sentiment Analysis

```
> UNSent <- analyzeSentiment(UNC)

> summary(UNSent)

   WordCount      SentimentGI       NegativityGI      PositivityGI    SentimentHE
 Min.   :   41   Min.   :-0.0654   Min.   :0.00166   Min.   :0.005   Min.   :-0.0112
 1st Qu.:  703   1st Qu.: 0.0830   1st Qu.:0.11518   1st Qu.:0.230   1st Qu.: 0.0113
 Median : 1193   Median : 0.1128   Median :0.13433   Median :0.248   Median : 0.0170
 Mean   : 1299   Mean   : 0.1127   Mean   :0.13460   Mean   :0.247   Mean   : 0.0172
 3rd Qu.: 1747   3rd Qu.: 0.1430   3rd Qu.:0.15441   3rd Qu.:0.266   3rd Qu.: 0.0224
 Max.   :10306   Max.   : 0.2745   Max.   :0.23671   Max.   :0.358   Max.   : 0.0724

  NegativityHE      PositivityHE     SentimentLM       NegativityLM     PositivityLM
 Min.   :0.00000   Min.   :0.0003   Min.   :-0.1188   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.00435   1st Qu.:0.0195   1st Qu.:-0.0432   1st Qu.:0.0451   1st Qu.:0.0262
 Median :0.00702   Median :0.0241   Median :-0.0239   Median :0.0571   Median :0.0319
 Mean   :0.00746   Mean   :0.0247   Mean   :-0.0271   Mean   :0.0595   Mean   :0.0324
 3rd Qu.:0.01011   3rd Qu.:0.0291   3rd Qu.:-0.0092   3rd Qu.:0.0732   3rd Qu.:0.0385
 Max.   :0.02494   Max.   :0.0724   Max.   : 0.0440   Max.   :0.1360   Max.   :0.0677

 RatioUncertaintyLM SentimentQDAP     NegativityQDAP   PositivityQDAP
 Min.   :0.0000     Min.   :-0.0660   Min.   :0.0000   Min.   :0.00331
 1st Qu.:0.0108     1st Qu.: 0.0636   1st Qu.:0.0561   1st Qu.:0.14384
 Median :0.0143     Median : 0.0843   Median :0.0746   Median :0.16030
 Mean   :0.0148     Mean   : 0.0844   Mean   :0.0763   Mean   :0.16074
 3rd Qu.:0.0185     3rd Qu.: 0.1079   3rd Qu.:0.0942   3rd Qu.:0.17779
 Max.   :0.0440     Max.   : 0.2308   Max.   :0.1741   Max.   :0.26003
```

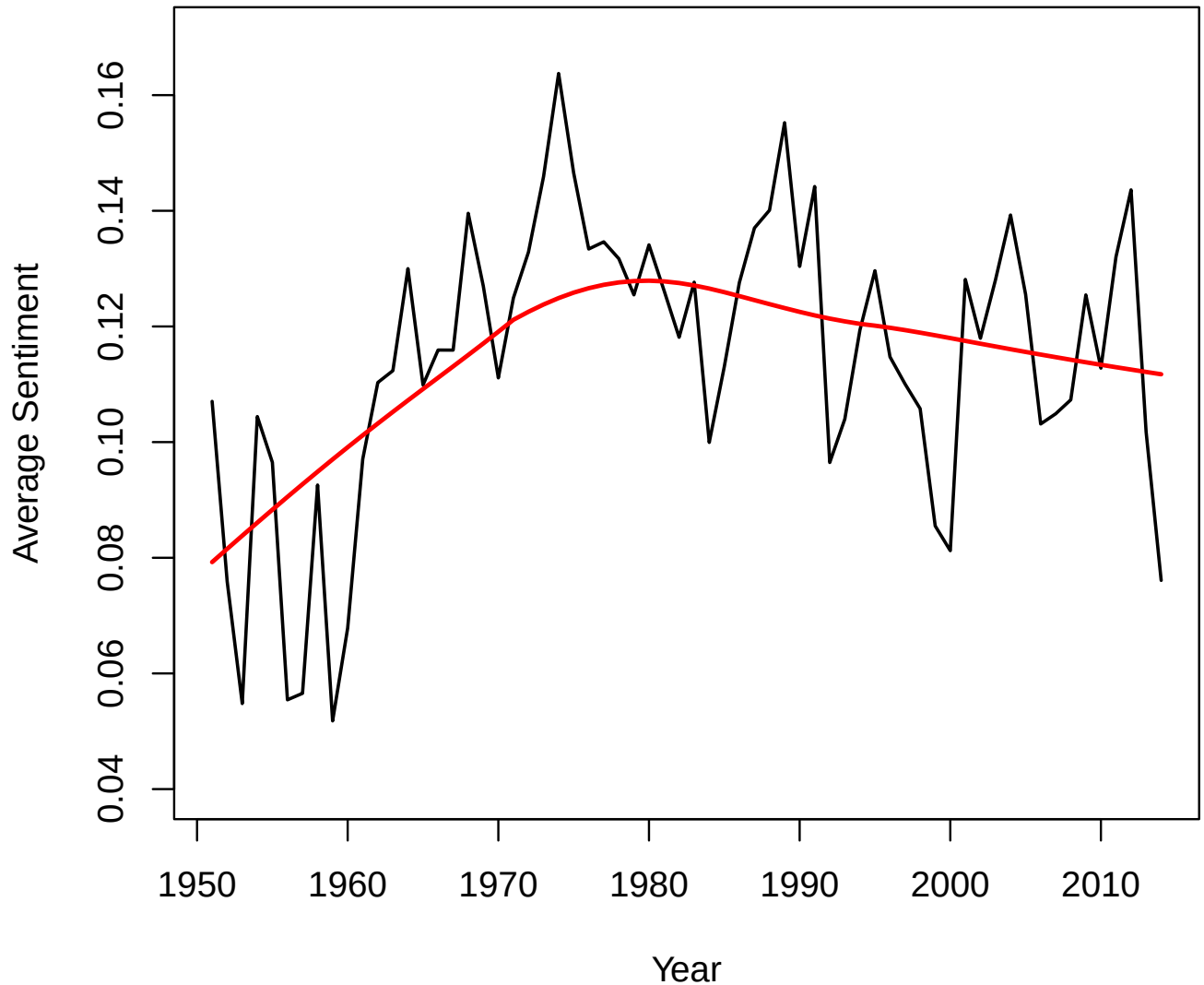
UNHCR: Sentiment vs. Word Count



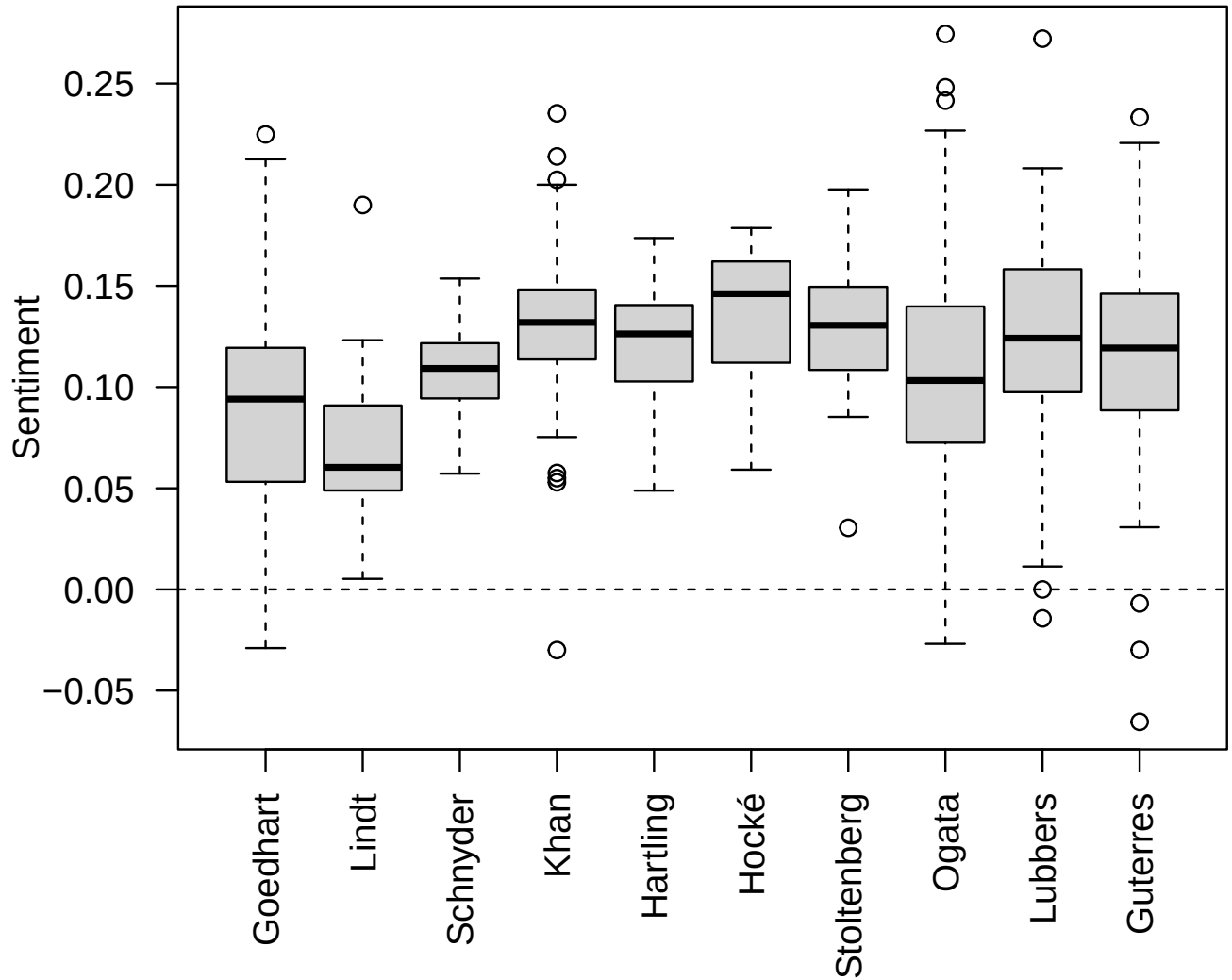
UNHCR: Sentiment Over Time



UNHCR: Annual Sentiment Means



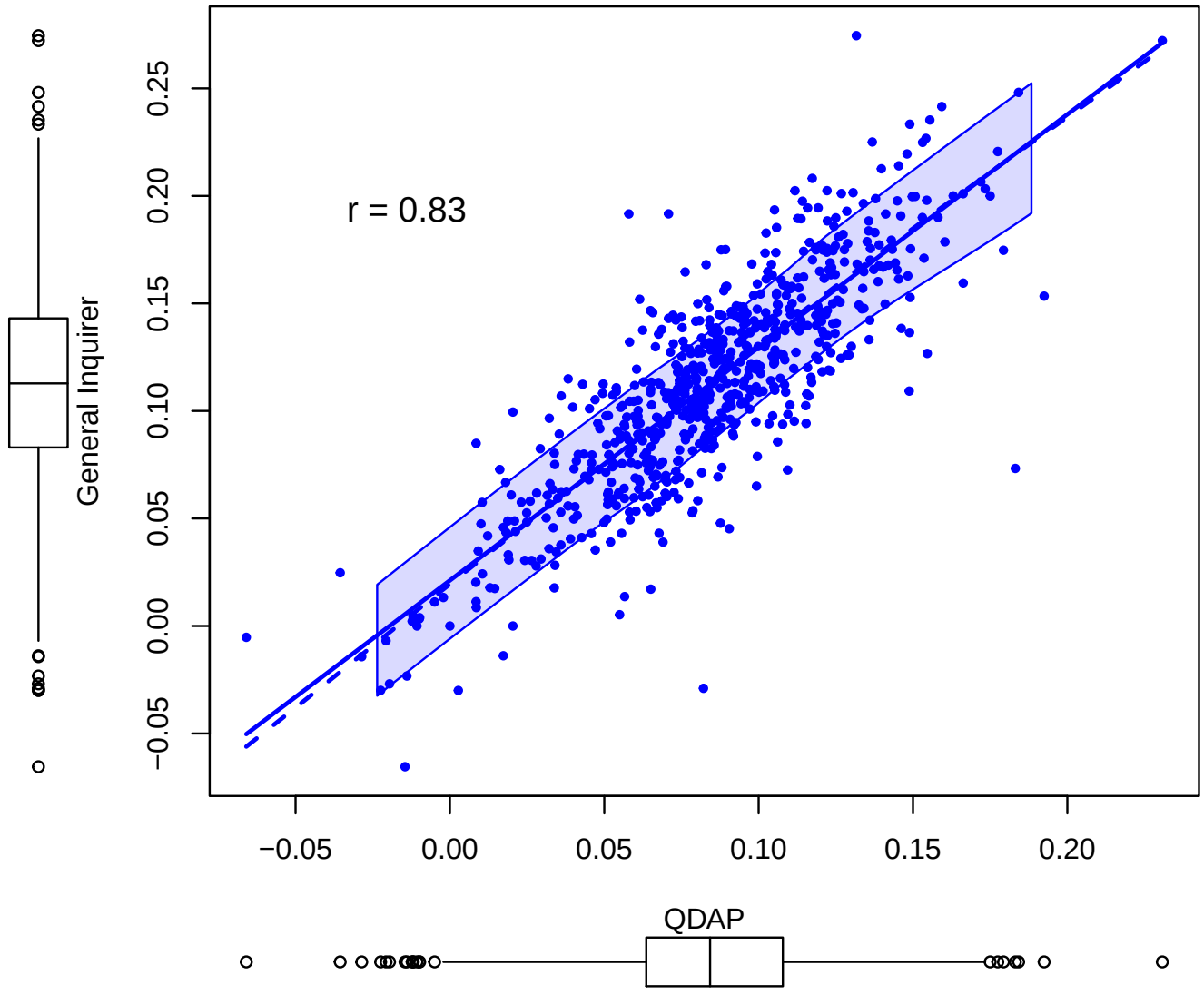
UNHCR: Sentiment By Speaker



Similar Results By Dictionary?

```
> GI<-loadDictionaryGI()
> QD<-loadDictionaryQDAP()
>
> compareDictionaries(GI,QD)
Comparing: binary vs binary

Total unique words: 5100
Matching entries: 2136 (0.42%)
Entries with same classification: 1448 (0.28%)
Entries with different classification: 63 (0.012%)
$totalUniqueWords
[1] 5100

$totalSameWords
[1] 2136

$ratioSameWords
[1] 0.42

$numWordsEqualClass
[1] 1448

$numWordsDifferentClass
[1] 63

$ratioWordsEqualClass
[1] 0.28

$ratioWordsDifferentClass
[1] 0.012
```

Comparing Results w/Different Dictionaries



Creating Custom Dictionaries “By Hand”

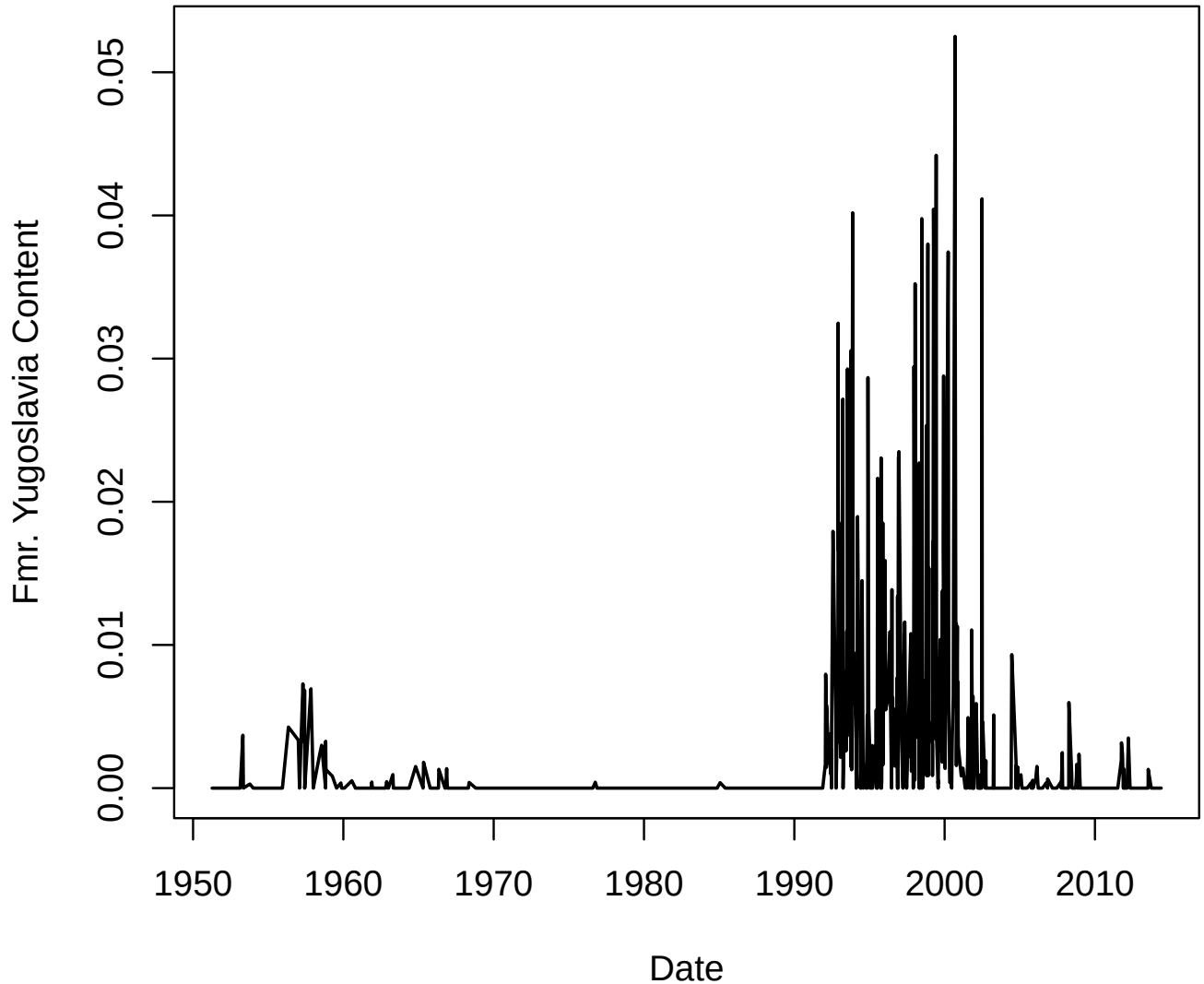
- Conflict in the former Yugoslavia, 1991-1999
- $\approx$ 2.3 million refugees
- “Europe’s biggest refugee crisis since World War II”
- Machine code speeches for content about the former Yugoslavia...

Create and Use a Custom Dictionary

Create a dictionary that keys on topical content (terms) related to the former Yugoslavia:

```
> YugoWords <- c("yugoslavia","serbia","bosnia","herzegovina",
+                "kosovo","montenegro","macedonia","croatia",
+                "vojvodina","balkans")

> FmrYugo <- SentimentDictionaryWordlist(YugoWords)

> UNHCRYugo <- analyzeSentiment(UNC,
+                               rules=list("YugoTopic"=list(
+                                 ruleRatio,FmrYugo)))


> summary(UNHCRYugo$YugoTopic)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.00000 0.00000 0.00000 0.00319 0.00334 0.05251
```

“Former Yugoslavia” Scores Over Time



Extensions / Challenges / etc.

Other miscellanea:

- **Linguistic complexity**
  - Irony, sarcasm, tone, etc.
  - Complex / subtle negation ("I don't have one guitar; I have many.")
- **Dictionaries**...
  - Specialized vocabularies $\rightarrow$ standard sentiment dictionaries break down (e.g., "love" in tennis)
  - *Minimally-supervised* dictionary creation (Rice & Zorn)
  - Bleeding edge: *Unsupervised* dictionary creation via negations...
- **Change over time**
  - Word meanings...
  - Word usage...

Topic Models

Topics in Text

- "Topics" / "themes" / etc.: What the document is *about*.
- How do we know?
  - Word meanings...
  - *Clustering* of words
  - Tone (sometimes)
- Complications / challenges...
  - What's a "topic"?
  - (Key)words can be ambiguous ("tennis" vs. "crane")
  - Documents are often about $>$ one topic

Extracting Topics

Dictionary-based / Supervised methods

- A la sentiment analysis...
- Predetermined "topics" (think: dictionaries of keywords)
- Topic$_i$ $\rightsquigarrow$ whatever topic(s) have (proportionally) the most terms

Unsupervised methods

- Extract topics from the corpus itself
- Intuition: *co-occurrence* of terms in documents
- Useful when (a) we don't know topics *a priori*, and/or (b) term meaning/usage is complex / nonstandard

Latent Dirichlet Allocation ("LDA")

Intuition:

- Start with N documents $i \in \{1...N\}$ in a corpus
  - Each document i has M_i total words
  - The total of all words in the corpus is V
- Each document comprises a *mixture* of one or more of k topics
- Each topic comprises a *mixture* of terms
- We observe documents and terms, but not topics; topics are *latent*
- *Goals*:
  - Infer the latent topic structure of the corpus
  - Assign documents (probabilistically) to topics
- *Process*:
  - Assign words to topics
  - Assess Pr(topic | document) and Pr(word | topic)
  - Reassign words to topic
  - Repeat...

Some options:

- `topicmodels` package
  - Plays well with `tm`
  - LDA and CTM estimation via VEM or Gibbs sampling
  - Some nice graphical tools
  - Is tidy-compatible (see here)
- `stm` package: Structural Topic Models
  - Fits the model in Roberts et al.
  - See the *vignette* / *website*
- Others (`quanteda`, `lda`, `text2vec`, `mscstexta4r`)

Example, Redux: UNHCR Speeches

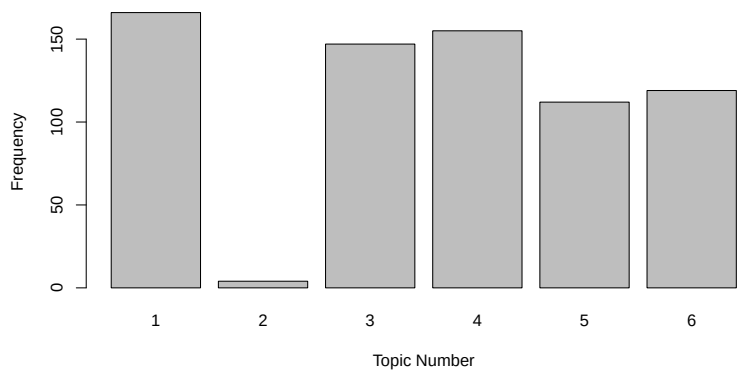
```
> UN <- read.csv("https://github.com/PrisonRodeo/OSU-Text/raw/main/Data/UNHCRSpeeches.csv")
>
> UN$Year <- as.numeric(str_sub(UN$by, -4)) # Year of the speech
> UN$foo <- str_extract(UN$by, '\\b[^,]+$')
> UN$Date <- as.Date(UN$foo, format="%d %B %Y") # date of speech
> UN$content <- gsub("\\\\n", " ", UN$content) # remove line breaks
> UN$foo <- NULL
> UN$Author <- "Goedhart"  # Fix names
> UN$Author <- ifelse(UN$author=="lindt",paste("Lindt"),UN$Author)
.
.
.
> UN$Author <- ifelse(UN$author=="guterres",paste("Guterres"),UN$Author)
>
> UNHCR <- textProcessor(UN$content, metadata=UN)
Building corpus...
Converting to Lower Case...
Removing punctuation...
Removing stopwords...
Removing numbers...
Stemming...
Creating Output...

> UNCorp <- prepDocuments(UNHCR$documents,UNHCR$vocab,UNHCR$meta)
Removing 6617 of 15688 terms (6617 of 403368 tokens) due to frequency
Your corpus now has 703 documents, 9071 terms and 396751 tokens.

> UNLDACorp <- convertCorpus(UNCorp$documents,UNCorp$vocab,
+                            type="slam")
```

Fit a Standard LDA

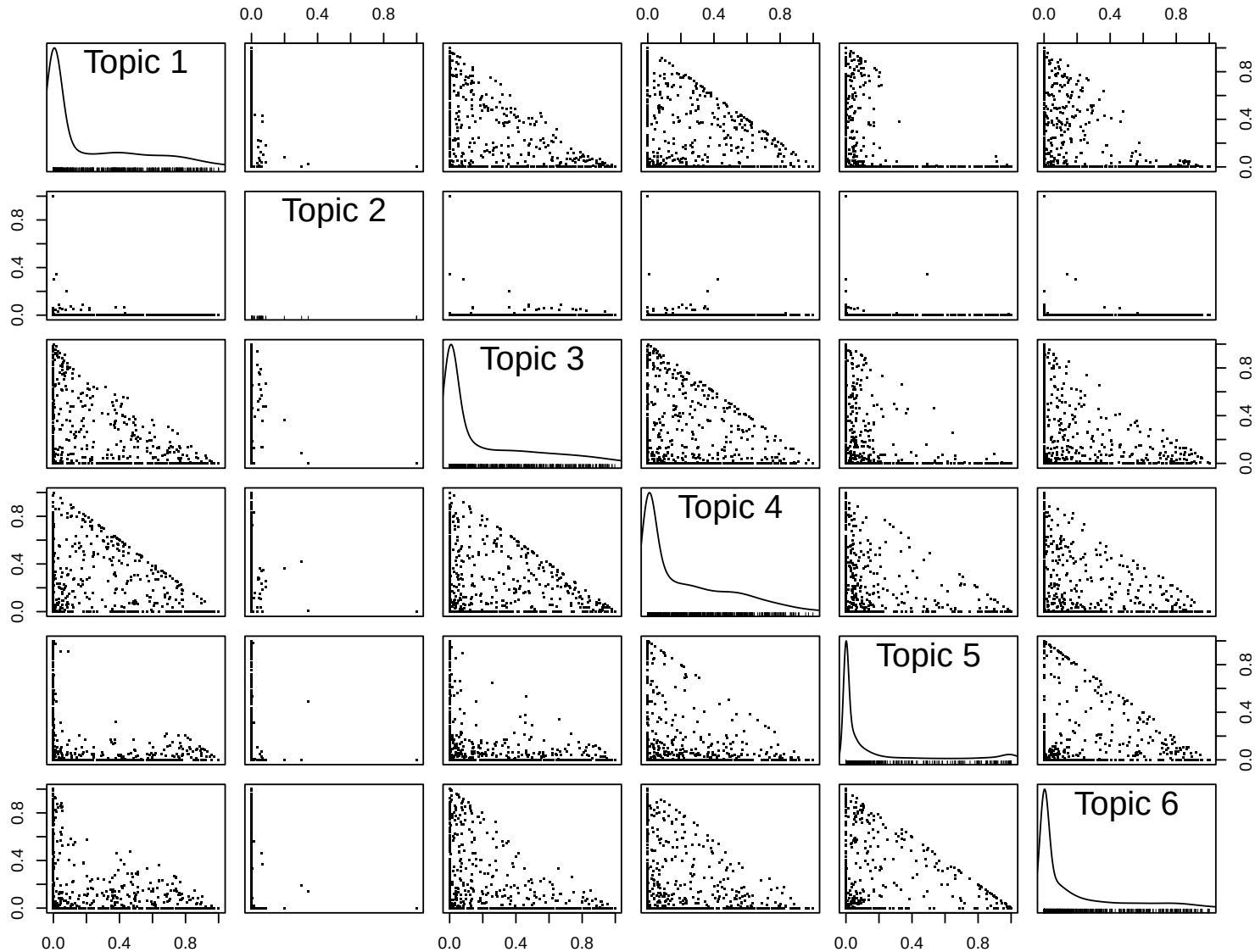
Model fitting:

```
> UN.LDAV.6 <- LDA(UNLDACorp,6,method="VEM",
+                  seed=7222009)

> # Check out the terms / topics:
>
> terms(UN.LDAV.6,10)
      Topic 1        Topic 2     Topic 3   Topic 4   Topic 5      Topic 6
 [1,] "humanitarian" "los"       "refuge"  "refuge"  "refuge"     "refuge"
 [2,] "refuge"       "que"       "unhcr"   "intern"  "problem"    "unhcr"
 [3,] "return"       "las"       "will"    "countri" "offic"      "assist"
 [4,] "displac"      "refugiado" "need"    "protect" "countri"    "govern"
 [5,] "intern"       "syrian"    "protect" "peopl"   "will"       "programm"
 [6,] "unhcr"        "para"      "year"    "right"   "govern"     "countri"
 [7,] "conflict"     "por"       "intern"  "human"   "year"       "nation"
 [8,] "secur"        "syria"     "countri" "asylum"  "high"       "unit"
 [9,] "will"         "una"       "develop" "state"   "programm"   "problem"
[10,] "peac"         "del"       "also"    "nation"  "commission" "will"
```

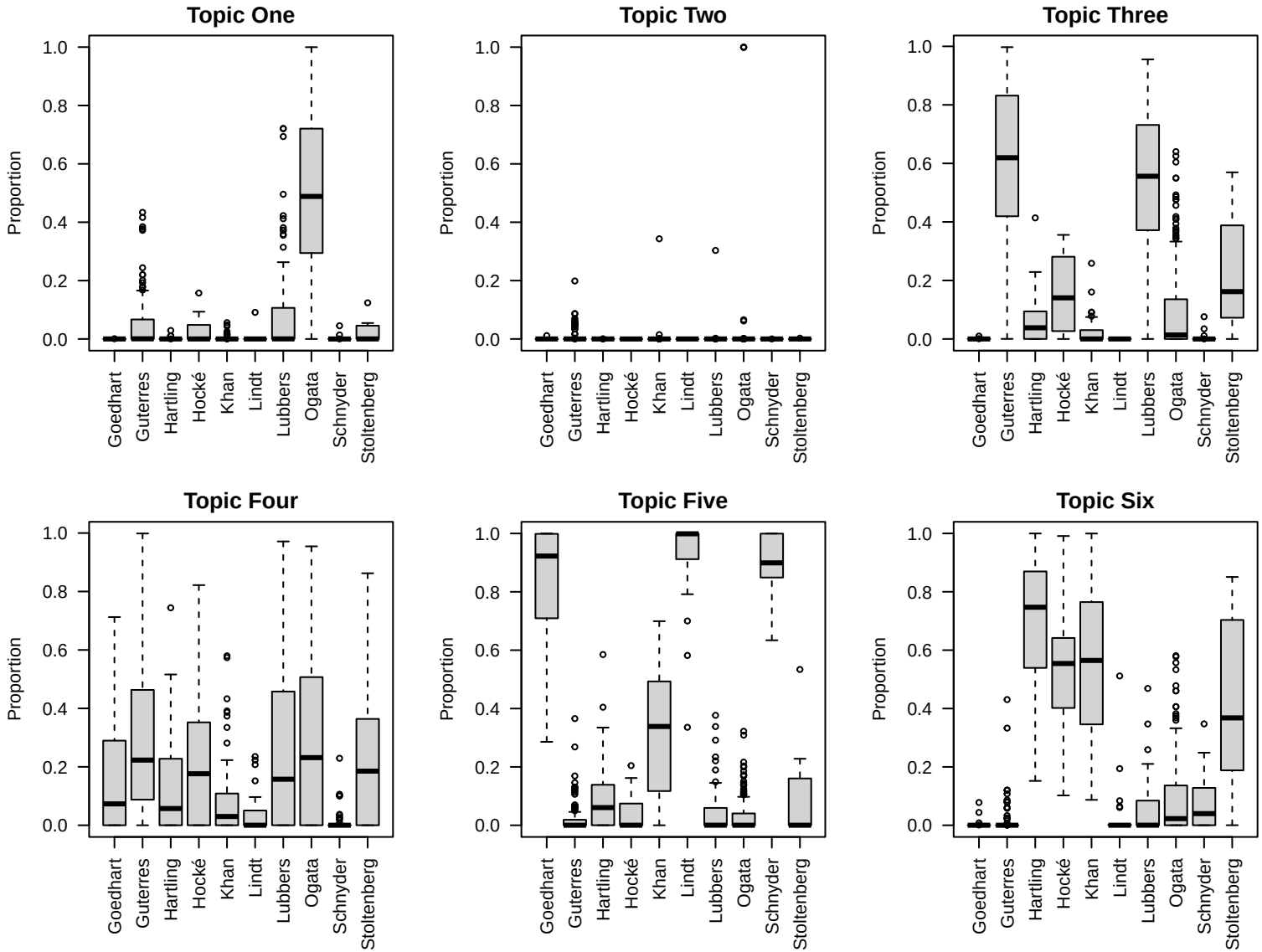
Document distribution across topics:



Posterior Topic Probabilities



Topic Probabilities by Author



Something to Think About: How Many Topics?

From the `stm` documentation:

> *"The most important user input in parametric topic models is the number of topics. There is no right answer to the appropriate number of topics.* **More topics will give more fine-grained representations of the data at the potential cost of being less precisely estimated.** *The number must be at least 2 which is equivalent to a unidimensional scaling model. For short corpora focused on very specific subject matter (such as survey experiments) 3-10 topics is a useful starting range. For small corpora (a few hundred to a few thousand) 5-50 topics is a good place to start. Beyond these rough guidelines it is application specific. Previous applications in political science with medium sized corpora (10k to 100k documents) have found 60-100 topics to work well. For larger corpora 100 topics is a useful default size. Of course, your mileage may vary." (emphasis added)*

Text Scaling

Scaling, in general:

- Goal: Combine/aggregate information (data reduction)
- Underlying assumptions...
    - Individuals speaking/writing/etc. differ in systematic, measurable ways
    - Those differences manifest themselves in text...
        - *What* they say
        - *When* they say it (topic selection)
        - *How* they say it (style, tone, etc.)
    - The mapping from latent differences to text is *systematic* and *observable*, and
    - Can be learned via analysis of the text itself

Intuition: Go from word frequencies / co-occurrences to measures of latent phenomena.

Supervised Text Scoring

Basic idea:

1. We know some documents' / authors' locations
2. Assess which terms in those documents give it it's location (distinctive)
3. Use the resulting term-level scores to locate other documents

One example: "Wordscores" (originally for scoring legislative text: speeches, press releases, etc.)

Unsupervised Text Scoring

Basic idea:

1. Assume that words $\mathbf{X}$ are generated according to some PDF $f(\cdot)$, with (latent) parameters θ for the units being scaled
2. Assess $\Pr(\theta|f(\cdot), \mathbf{X})$
3. Resulting posterior $\hat{\theta}$ are estimates of the parameters + variability
4. Also: Estimates of ideological *clarity* / *ambiguity* (Lo, Proksch and Slapin 2014)

Characteristics:

- IRT-like...
- One example: "Wordfish" (Slapin and Prokschk 2008) (also originally for scoring legislators)

Text Scaling: Options in R

A few choices:

- `quanteda` (Benoit et al.)
  - Code for supervised (Wordscores, the "class affinity model," linear SVM + logistic) and unsupervised (Wordfish, correspondence analysis, latent semantic analysis) models
  - Also includes different suites of packages for `textmodels`, `textstats`, and `textplots`
  - Well-documented and supported
- R packages for correspondence analysis, naive Bayes classification of text, etc.
- Various others (e.g., Slapin's `wordfish` code)

In practice, **they're all pretty much the same**.

NLP: Transformers and LLMs

Going Bagless

"Bag of words" models are

- Simple
- Fast ($\longrightarrow$ scalable)
- Flexible

They are also:

- Entirely devoid of context
- $\longrightarrow$ *Terrible* at semantics
- Poor at learning from short documents...

Context → Embeddings

Embeddings contextualize words...

- AKA "distributional semantics," "distributional methods," "word embeddings," "word vectors," etc.
- Each word (token) is represented by an n-dimensional *vector*
- The value for each dimension is determined by which words are commonly found next to / near that word in a k-sized window
- Exist at the *word*-level (generally)
- For more, see (e.g.) here, or here, or here (if you have a bit of Python background)

Classic example:

"Queen" = **"King"** - **"Man"** + **"Woman"**

An even better example (via Michael Burnham):

	Animal	Fluffy	Pet	Dangerous	Food
Cat	.98	.94	.87	.30	.15
Dog	.98	.87	.95	.35	.15
Pig	.95	.20	.45	.26	.60
Carrot	.01	.02	.01	.01	.97
Rock	.01	.02	.43	.25	.01

Embeddings: How Do They Work?

Intuition:

- Basis in neural network models...
  - "One-hot" representations: For (e.g.) "The cat ate the rat":

    ```
    the = [1 0 0 1 0]
    cat = [0 1 0 0 0]
    etc.
    ```

  - This is *inefficient*; simpler to represent each word with a vector of `associations`

- An early example was `Word2Vec`, which used the cosine similarity of vectors for word associations

- Embeddings models are trained on (very) large datasets (e.g., the entire internet)

The Good, The Bad

Text tools based on embeddings:

- ...are a far more accurate & robust model of language
- Work excellently with short texts
- Eliminate the need for text pre-processing

They are also:

- *Very* computationally demanding
- Not especially intuitive or flexible
- Effectively “black boxes”

Context: The Problem With Cats

Consider:

"This is my cat Miles."

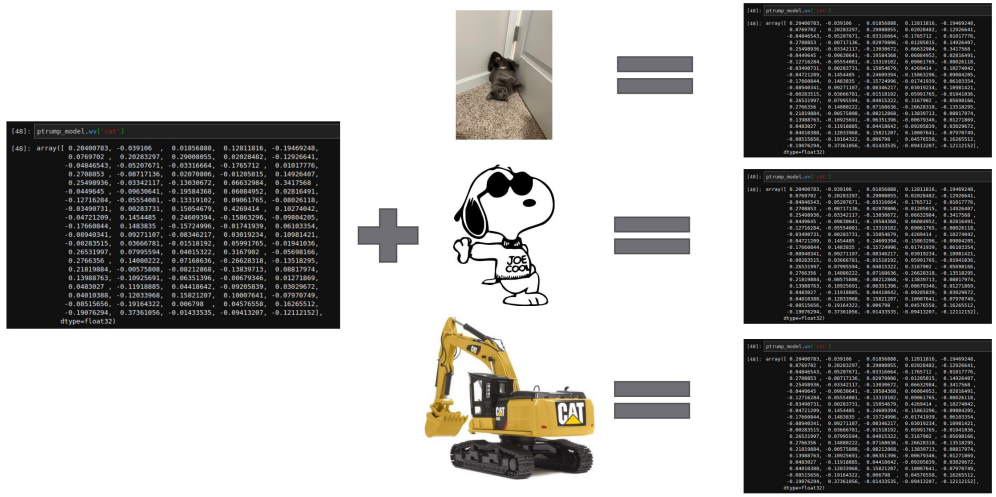
"Who let the cat out of the bag?"

"Dr. Steel is one cool cat."

"That CAT dug the trench in no time."

"It's raining cats and dogs."

Global embeddings → Contextual Embeddings...



(source)

Contextual Embeddings

How?

- **Foundation Models**: Extensive models trained on very large corpuses, intended for downstream use.

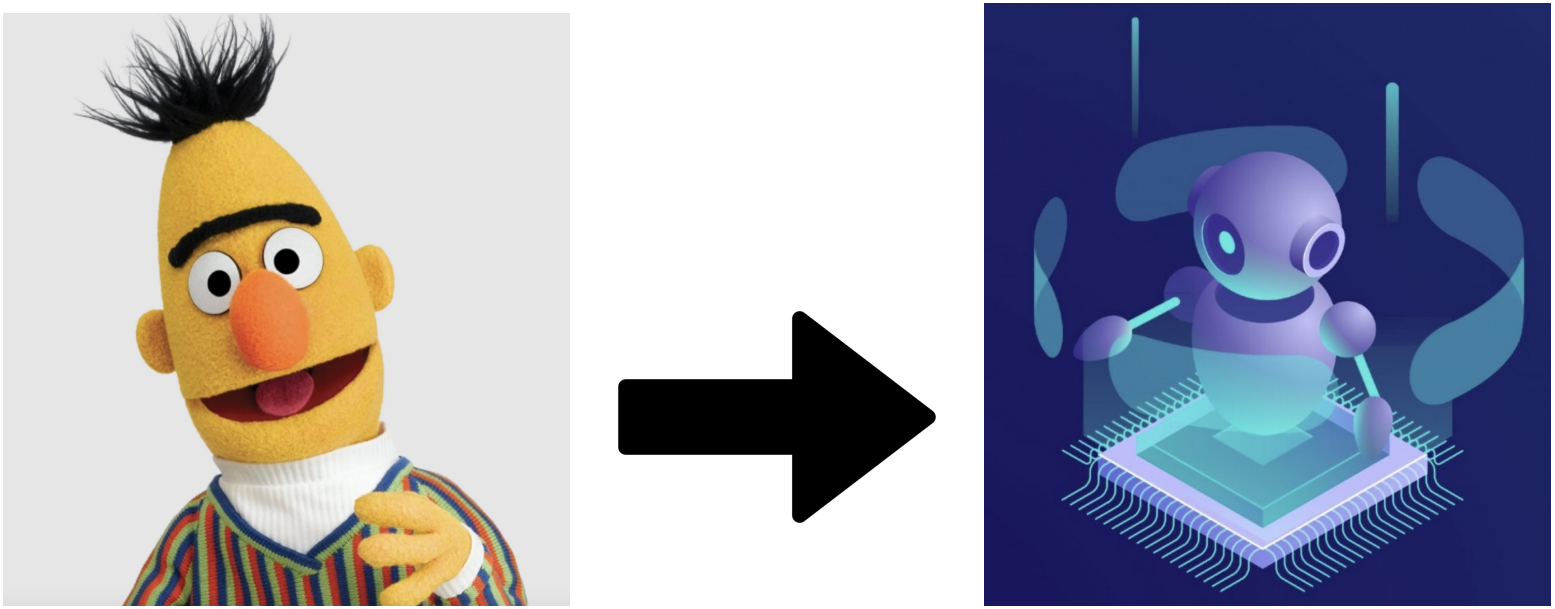
$$+$$

- **Transfer Learning**: Taking knowledge from one model and applying it to another.

Example: **Bidirectional Encoder Representations from Transformers (BERT)**

- A *transformer network* – self-attentive learning model
- Uses contextualized embeddings
- Applications: Supervised and unsupervised classification, topic models, sentiment analysis, translation, document similarity, search / question answering, etc.

BERT → GPT



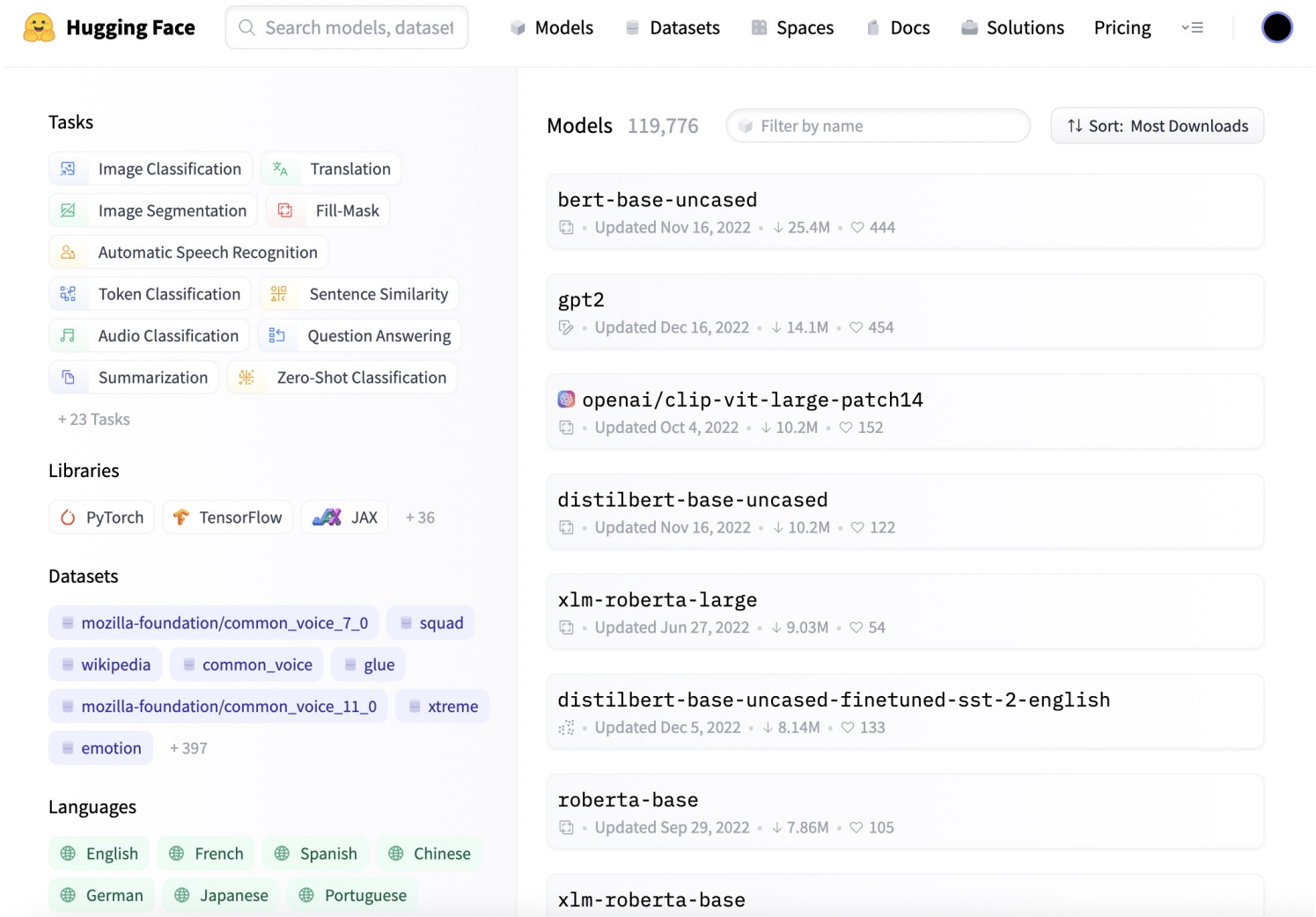
Generative Pretrained Transformer (GPT

- Larger version of BERT, trained on more data
- Primary geared toward text / natural language generation
- E.g. ChatGPT (https://chat.openai.com)

A few tips:

- **Software**
  - Python > R (but see below...)
  - Work within *virtual (machine) environments* – e.g., via Conda
- **Hardware**
  - Local: Requires and NVIDIA GPU – *or*
  - You can access all the GPU you need through a free account on the Google Colaboratory (Python only)
- **Resources**
  - A good start: Michael Burnham's stance detection tutorial
  - The place where it all happens in Hugging Face 🤗

Hugging Face



A Very Simple Example: Sentiment

Note that making this example[1] work (here, on a recent Mac) requires:

- Installing Python (language)
- Installing Anaconda and/or Pip (package managers – I use Pip here)
- Using Pip to install the Pytorch package

How each of these things is accomplished varies depending on many factors (operating system, etc.); follow the links in the examples for details...

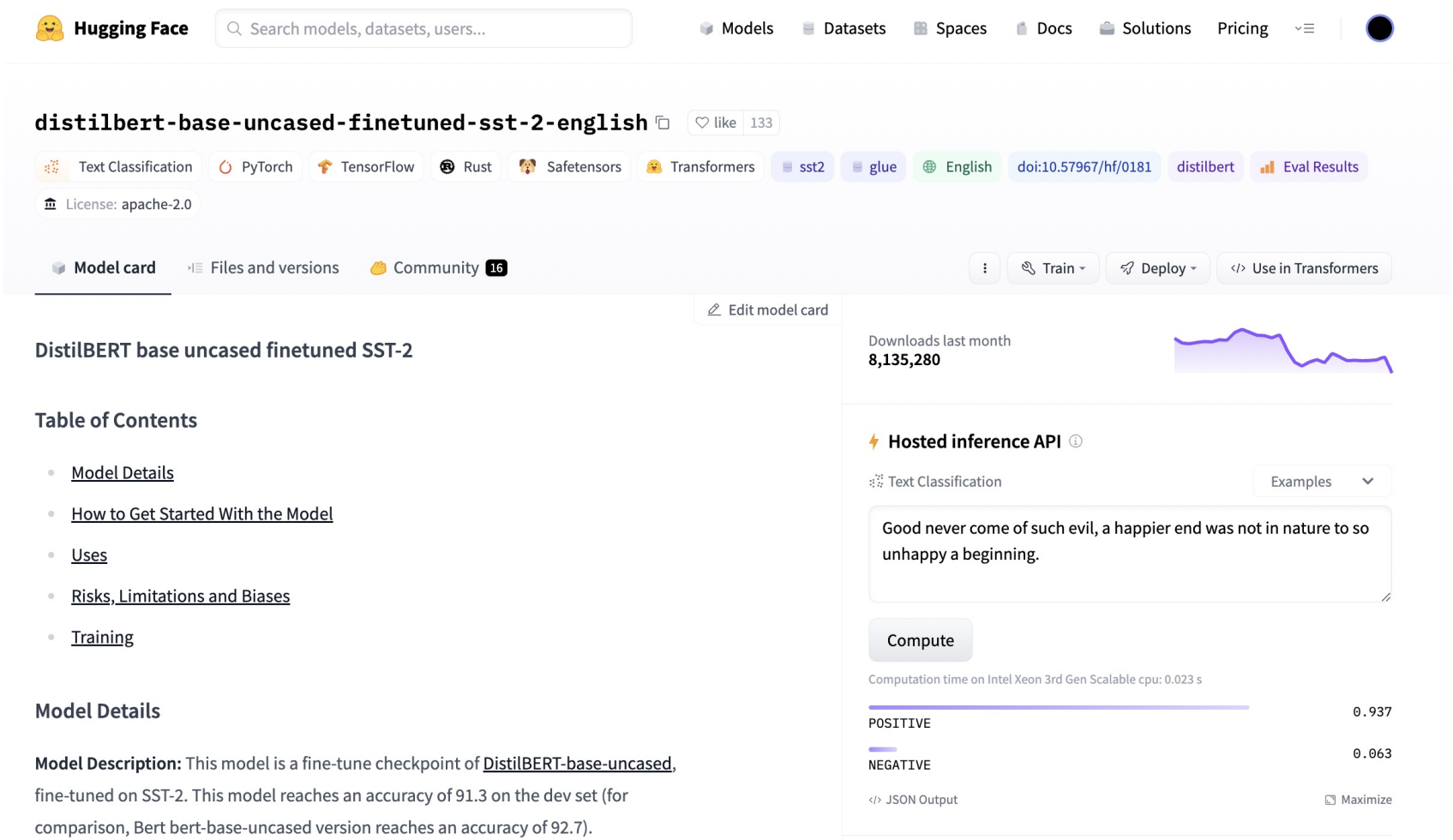
[1] Stolen, shamelessly but with attribution, from this blog post.

A Very Simple Example (continued)

Once we have that, we can use commands from the `reticulate` package to call Python commands in R:

```
# Start Python:

use_python("/Users/cuz10/foo/bin/python")
py_config()

# ... then install the transformers:

py_install("transformers",pip=TRUE)

# ...and bring them into the current R session:

> trans<-reticulate::import("transformers")

# Then create the "pipeline" through which you'll feed your text:

> sentimentGPT<-trans$pipeline(task="text-classification")

No model was supplied, defaulted to distilbert-base-uncased-finetuned-sst-2-english and revision
af0f99b (https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english).
Using a pipeline without specifying a model name and revision in production is not recommended.
```

The BERT Model We're Using



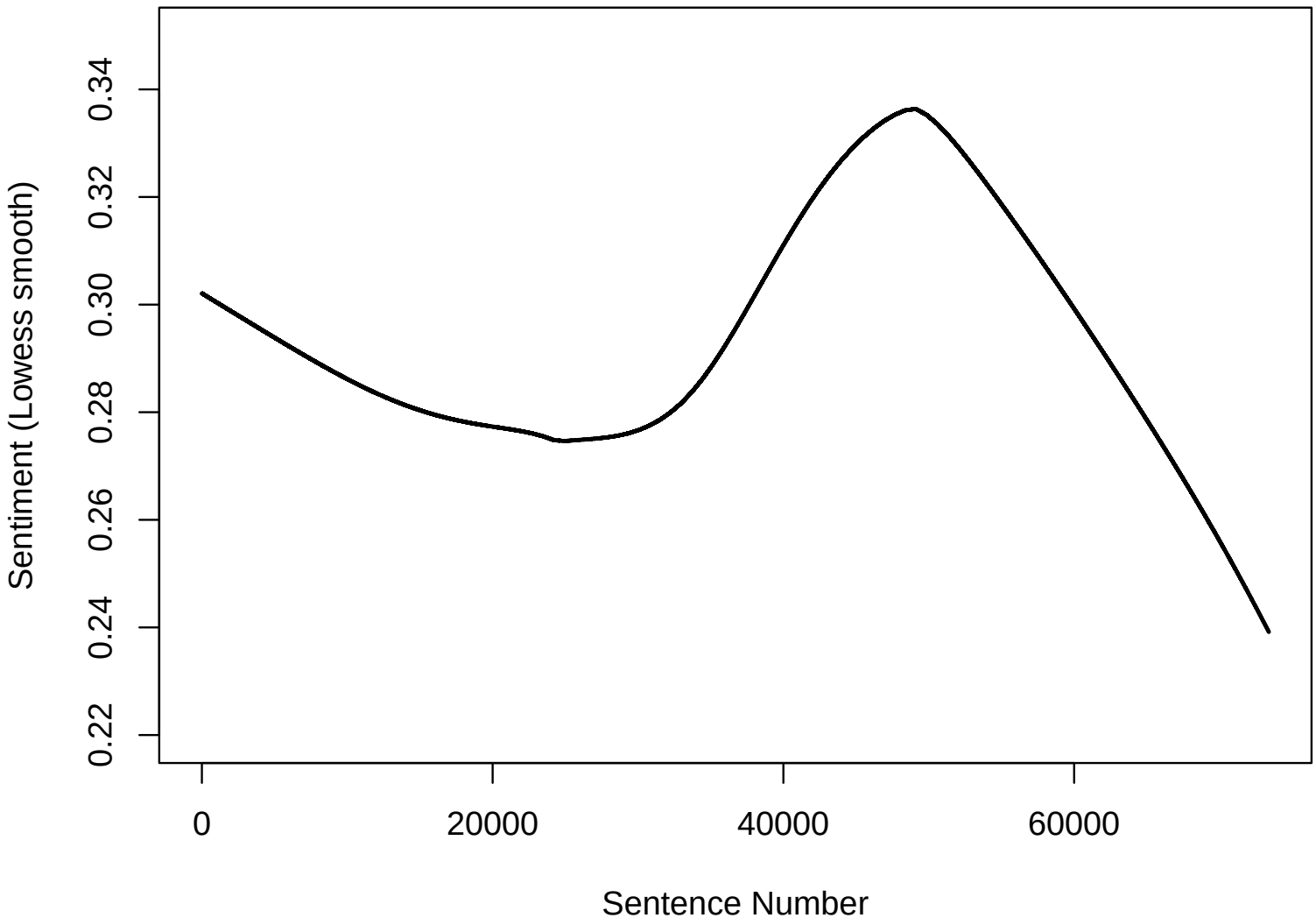
A Very Simple Example (continued)

Pass some text (here, the Gettysburg Address) through the "pipeline" to generate its sentiment score:

```
> GBA.sentiment<-sentimentGPT(GBA)
> GBA.sentiment
[[1]]
[[1]]$label
[1] "POSITIVE"

[[1]]$score
[1] 0.9857
```

UNHCR Sentiment Scores Via Transformers

Do the same for all 73,394 sentences in the UNHCR Speech data:



A Concluding Frame

Lessons:

- Immediate usage: measurement (especially for bag-of-words models)
- Transformer models: Potential to do everything bag-of-words models do and more
- Current example: Argyle et al. (2022): "Out of One, Many: Using Language Models to Simulate Human Samples."
- Challenges:
  - Data acquisition
  - Tool development / ease of use
  - Collaboration (to find interesting problems)

Thank you!