Assessing the Consequences of Text Preprocessing Decisions

Matthew J. Denny[1]   Arthur Spirling

Penn State University   New York University

October 15, 20016

[1] Work supported by NSF Grant: DGE-1144860

Common Preprocessing Decisions

- P – **Punctuation Removal**
- N – **Number Removal**
- L – **Lowercasing**
- S – **Stemming**
- W – **Stopword Removal**
- I – **Infrequent Term Removal**
- '3' – **n-gram Inclusion**

7 binary choices $\longrightarrow 2^7 = 128$ specifications.

Supervised Learning

Unsupervised Learning

What Could Possibly Go Wrong?

Motivating Example

- UK Manifestos Corpus (1918–2001)
- Labour, Liberal, Conservative Parties
- Wordfish
  - Place documents in ideological space
- Process:
  1. Select preprocessing specification
  2. Run Wordfish

A-Priori Rankings

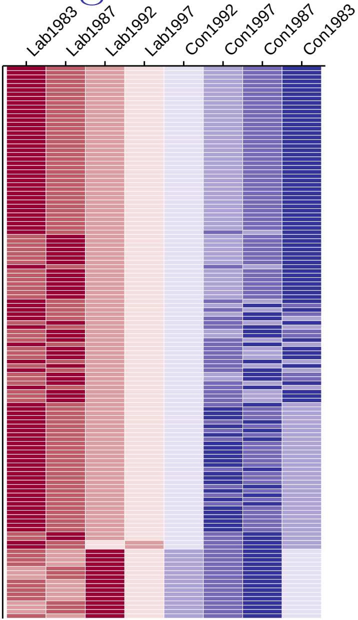
- Focus on 8 Manifestos.

  1. Four general elections (1983–1997)
  2. Labour and Conservative parties

- Lab $_{1983}$: "longest suicide note in history", extremely left–wing.

$$\boxed{\begin{array}{c}\text{Lab}_{1983} < \text{Lab}_{1987} < \text{Lab}_{1992} < \text{Lab}_{1997} < \\ \text{Con}_{1992} < \text{Con}_{1997} < \text{Con}_{1987} < \text{Con}_{1983}\end{array}}$$

Wordfish Rankings



Forking Paths

- 12 unique document rankings
- Substantially different conclusions.

Specification	Most Left	Most Right
P-N-S-W-I-3	Lab $_{1983}$	Cons $_{1983}$
N-S-W-3	Lab $_{1987}$	Cons $_{1987}$
N-L-3	Lab $_{1992}$	Cons $_{1987}$
N-L-S	Lab $_{1983}$	Cons $_{1992}$

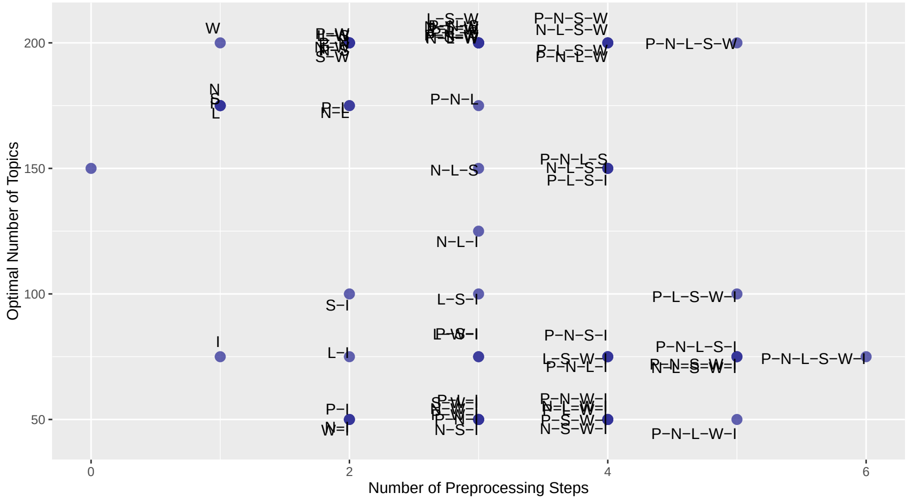
Another Example: Topic Models

- Senate Press Releases (Grimmer, 2010)
- Sample of 1,000 documents
  - 100×10 Senators.
- Note: no n-grams (computational cost).
- Procedure:
  1. Find optimal number of topics for each specification (perplexity).
  2. Run topic model (LDA)

Perplexity to Select Number of Topics

- Split data into train/test sets (80/20).
- Find minimum *perplexity* over num. topics.
- topics $= \{25, 50, 75, 100, 125, 150, 175, 200\}$
- 10-fold cross validation.

Optimal Number of Topics

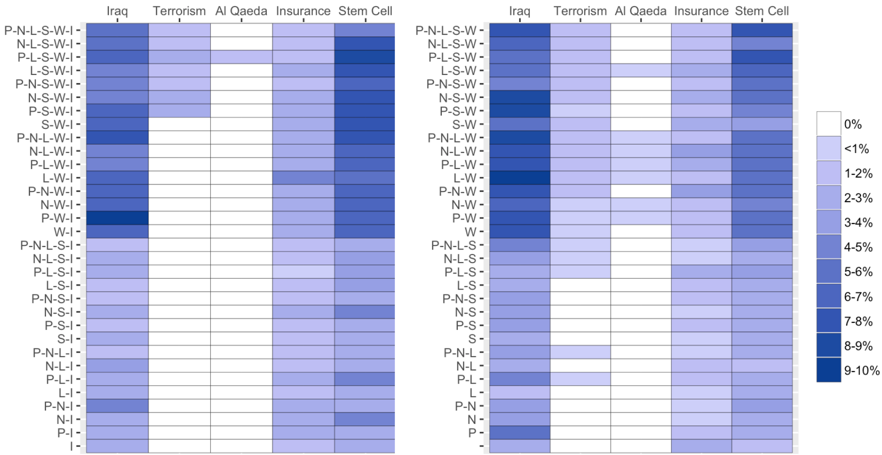


Key Terms Example

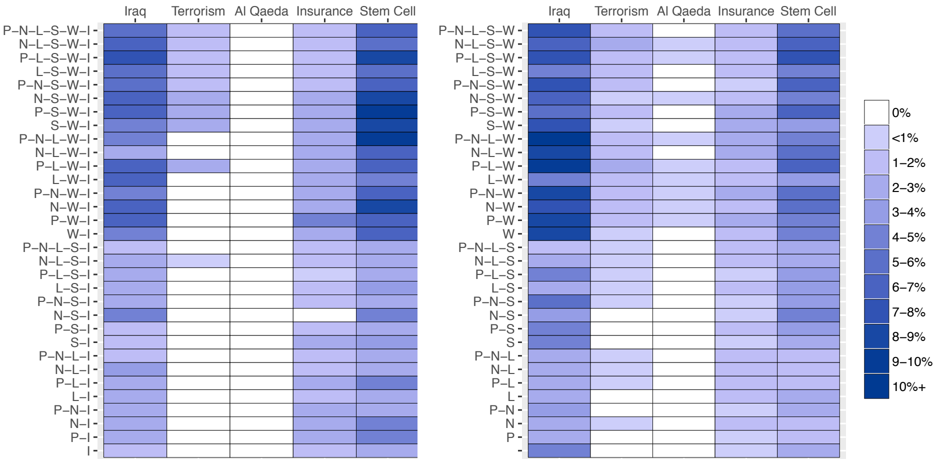
- Select five "key terms".
- How many topic top-terms are they in?

iraq
terror(ism)
(al) **qaeda**
insur(ance)
stem (cell)

Key Terms in Topic Top-Terms



Key Terms: Average of 40 Initializations



Forking Paths

- Different preprocessing $\longrightarrow$ different conclusions.
- Are we **doomed**?

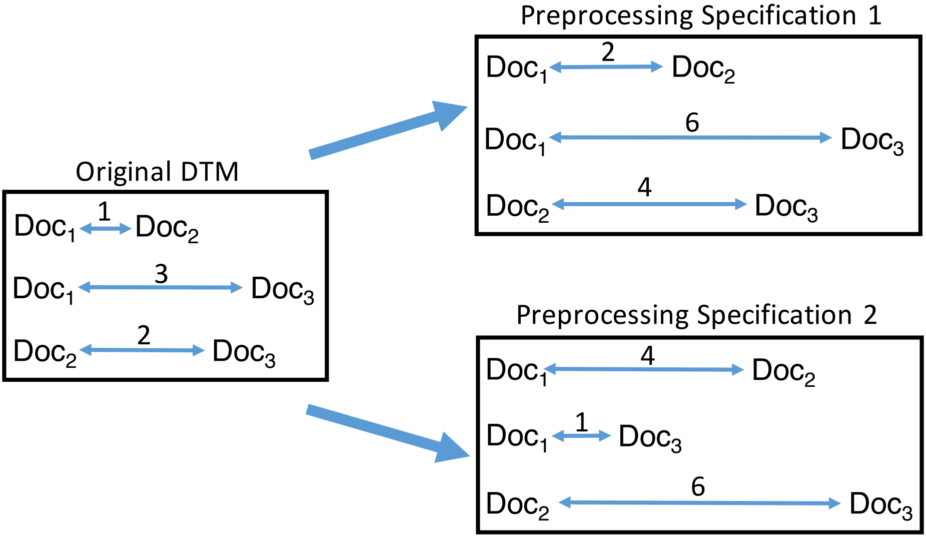
Our Solution: `preText`

- Assess consequences of preprocessing choices.
- Characterize a number of corpora.
- Easy to use `R` package!

Overview: Movements in Pairwise Document Distances

- No preprocessing as base case.
- Compare how **pairwise document distances** change with preprocessing.
- Measure how unusual these changes are.

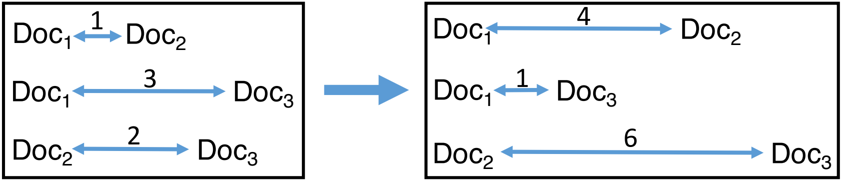
Example With Three Documents



Ranking Distance Changes

Original DTM

Preprocessing Specification 2



Original DTM	Preproc. Spec. 2	Abs. Difference
d(1,3) = 3	d(2,3) = 6	Δ d(1,3) = 2
d(2,3) = 2	d(1,2) = 4	Δ d(2,3) = 1
d(1,2) = 1	d(1,3) = 1	Δ d(1,2) = 1

Comparing Preprocessing Specifications

- Each specification will have a **largest mover**.

- Rank in other specifications $(M_1, ..., M_{127})$?

$$\mathbf{v_{M_1}} = (2_{M_2}, 14_{M_3}, 2_{M_4}, 3_{M_5}, \ldots, 15_{M_{127}}).$$

- Average of $\mathbf{v_{M_i}} \longrightarrow$ how **unusual**.

preText Scores

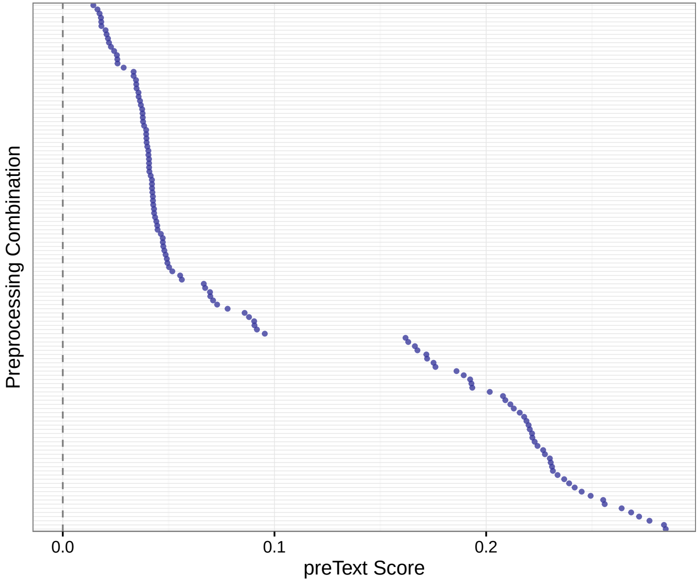
- Consider top k largest moving doc pairs.
- Average across $\mathbf{v_{M_i}} \longrightarrow \mathbf{v_{M_i}}^{(k)}$
- Normalize by $\frac{n(n-1)}{2}$ $(n = \text{num docs})$

$$\texttt{preText}\ \text{score}_i = \frac{2\mathbf{v_{M_i}}^{(k)}}{n(n-1)}$$

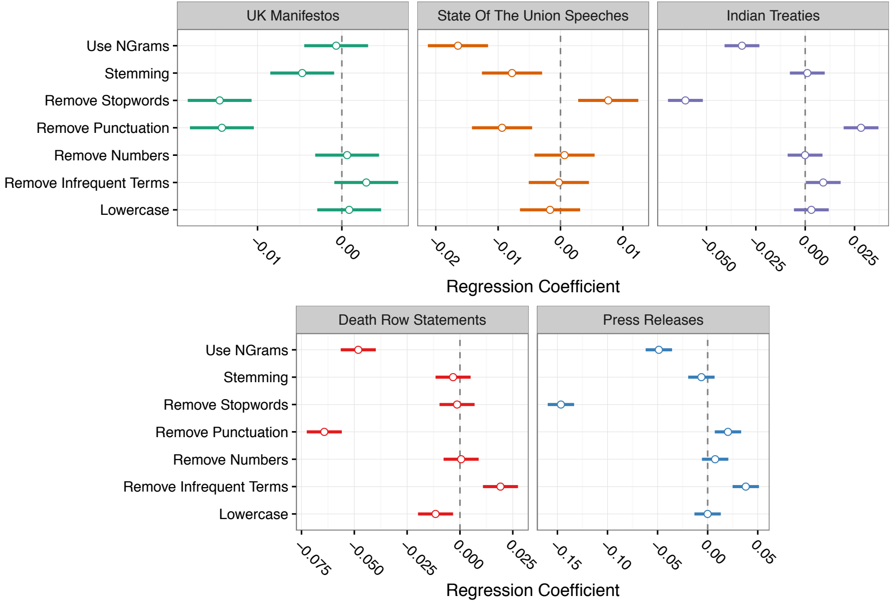
Interpreting `preText` Scores

- `preText` scores range between 0 and 1.
- **Lower** score $\longrightarrow$ **"typical"** changes in document distances.
- **Higher** score $\longrightarrow$ **"atypical"** changes in document distances.

`preText` Scores for Press Releases



Which Steps Matter?



Common Trends? (Danger!)

- Stopping, punctuation: highly variable.
- Stemming, numbers, lowercasing: not much effect.
- Including n-grams: potentially "good".
- Infrequent terms: potentially "bad".

Summary

- **Preprocessing matters.**
- **Forking paths** of inference.
- Our solution: `preText`.
- General Advice:
  - Some steps seem innocuous.
  - **Always check – tell reader!**

Happy Sloths Love R Packages!

- `install.packages("preText")`
- ssrn.com/abstract=2849145
- github.com/matthewjdenny/preText

Wordfish and `preText`