Automated Text Analysis in Political Science

Lecture 3: Preprocessing data, going from text to numbers

May 8, 2019

---

dr. Martijn Schoonvelde

School of Politics and International Relations, UCD

Today's class

- Implications of preprocessing text
  - Denny and Spirling (2018)
- Practice creating and visualizing a bag of words in **quanteda**
- Work on your own research design

Preprocessing text: theory

Preprocessing text: practice?

Denny and Spirling 2018

- From words to numbers
- Aim is to make "inputs to a given analysis less complex in a way that does not adversely affect the interpretability or substantive conclusions of the subsequent model" (Denny and Spirling 2018)
  - Simpler data, not too much information loss
- Standard in NLP and information retrieval, but does it make sense for political scientists as well?
  - In particular, when going from supervised to unsupervised methods
  - Goal is different: not classification but detecting latent structure
  - Supervised methods give us a clear benchmark, unsupervised methods do not

Denny and Spirling 2018

- "For just seven possible (binary) preprocessing steps, there would be $2^7 = 128$ possible models to run and analyze"
- Possibility of 'heading down "forking paths of inference" (Gelman and Loken 2014)
    - End result may crucially depend on arbitrary steps
- Denny and Spirling check this possibility for multiple datasets, and provide and R package that implements solutions
- Argument: preprocessing requires substantive knowledge + statistical check

Preprocessing steps

1. **P**unctuation
2. **N**umbers
3. **L**owercasing
4. **S**temming
5. **W** Stopword removal
6. **3** n-gram inclusion
7. **I**nfrequently Used Terms

NB: not all of these decisions are really binary; many possible stop word lists for example

Application 1: Wordfish on UK manifestos

- Wordfish model (Slapin and Proksch 2008) Labour and Conservative manifestos over 4 elections (1983 - 1997)
  - Estimates latent ideological position for each text
  - We'll see more of this model on Monday
- Prior belief of order of manifestos:

$$\text{Lab}_{1983} < \text{Lab}_{1987} < \text{Lab}_{1992} < \text{Lab}_{1997} < \text{Con}_{1992} < \text{Con}_{1997} < \text{Con}_{1987} < \text{Con}_{1983}.$$

Wordfish positions of UK manifestos

Figure 1. Wordfish results for the 128 different preprocessing possibilities. Each row of the plot represents a different specification. A white bar implies that the manifesto for that year is in the correct place as regards our priors. A black bar implies it was misplaced.

NB: twelve different orderings

Application 2: Topic models of Congress press releases

- LDA topic models (Blei, Ng & Jordan 2003) on congressional press releases
  - We'll learn more about this model (and extensions) early next week
  - Unsupervised method to learn about themes
- Main assumption: words in a text are generated from a set of topics

Topic models

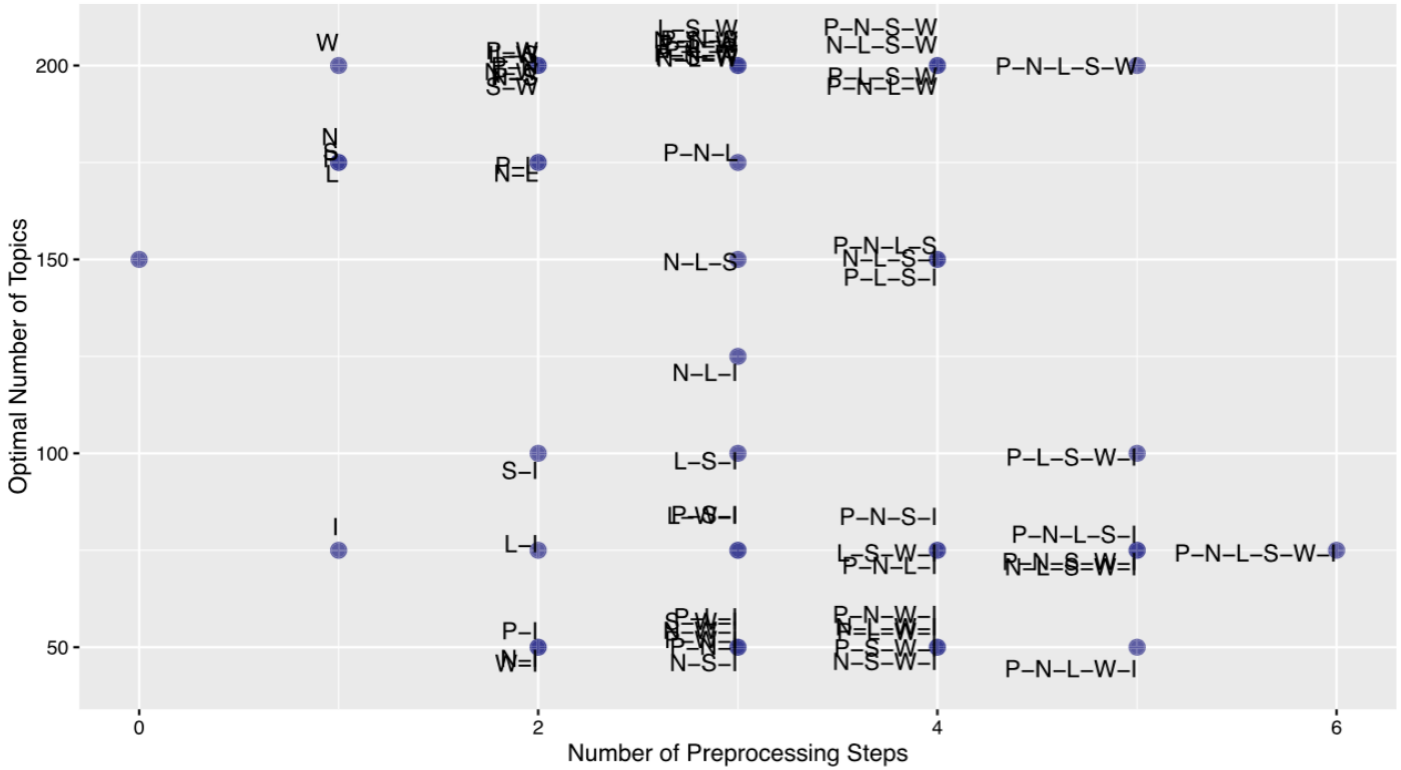

Figure 2. Plot depicting the optimal number of topics (as selected via perplexity) for each of 64 preprocessing specifications not including trigrams. On the x-axis is the number of preprocessing steps, and the y-axis is the number of topics. Each point is labeled according to its specification.

NB: "optimal" number of topics varies with preprocessing steps

Topic models

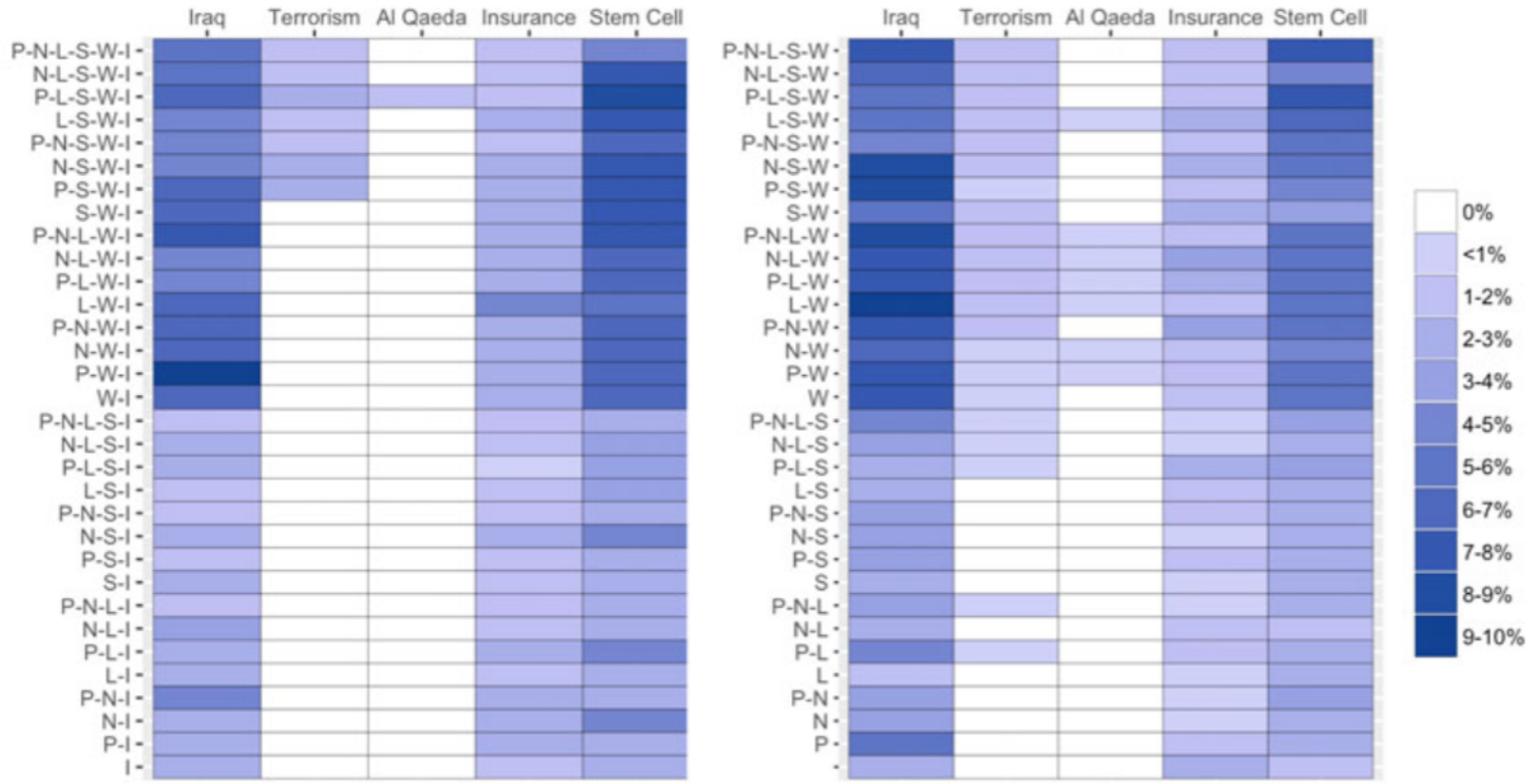

Figure 3. Plots depicting the percentage of topic top-20 terms which contain the stem of each of five keywords, for each of 64 preprocessing steps (thus excluding those which include trigrams). The number of topics for specifications fit to each of the 64 DFMs were determined through tenfold cross-validation, minimizing the model perplexity.

NB: topical content of documents varies with preprocessing steps

Pretext scores

- Use theory to guide your preprocessing
- Check sensitivity of outcomes to preprocessing steps:

  `Pretext`

  - Uses rank order of pairwise similarity / distance between documents
  - Takes mean difference rank order of a set of k documents from the corpus and divides by maximum possible rank differences
  - Summarized by a **Pretext** score (between 0 and 1)
    - More on similarity / distance tomorrow (cosine similarity and euclidian distance)

Practical recommendations

- `Pretext`
  - If Pretext scores do not vary with preprocessing, then results are robust
  - If Pretext scores do vary with preprocessing, evaluate outcomes against different preprocessing steps and strength of prior beliefs

Conclusion

- Pre-processing is not a neutral step in automated text analysis
- Denny and Spirling (2018) (see also Greene et al. (2016) find preprocessing steps matter
- But offer no theory on how and why they matter
    - Do preprocessing steps systematically influence unsupervised methods?

NOW FOR SOMETHING COMPLETELY DIFFERENT...

Coding Assignments and Research Note

- You'll use R Markdown for this – tool for dynamically reporting results with R
- Great integration with R Studio; documents compiled through Knitr
  - PDF; HTML; Word; Beamer; etc.
- Useful for combining code and text in one document

R Markdown in RStudio

- Let's take a look in R Markdown in Rstudio:
- Useful resource for formatting your research note:
  - http://rmarkdown.rstudio.com
  - `https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet`

Coding Assignments

- R Markdown (.Rmd) files provided for you
- Only a matter of filling in code chunks or answering in text boxes
- Plus change the metadata:
  - Title: "Assignment #"
  - Name: "First Name Last Name"
  - Document name: Assignment_#_First_Name_Last_Name
- Knit the .rmd document and send me both the PDF or HTML files

Research Note

1. Research question
2. Find texts
    - Develop the **corpus** to be analyzed
3. Clean text
    - Define **documents**, the unit of analysis
        - Paragraphs, documents, sentences
4. Preprocess text
5. Analyze text
    - What exactly do you want to extract from these texts?
    - This can be descriptive (lots of examples tomorrow), or model-based (from Friday onwards)
6. Visualize outcomes
7. Write up analysis and results

Research Design / Research Note

Open a R markdown file and start working on the following

1. What is your research question?
2. Develop expectations, building on work from others
    - Use Google Scholar to look for (empirical) papers that do something similar
3. What is your (empirically testable) expectation?
4. Describe your corpus. What are the units of observation? (That is, what are your documents?)
    - How many documents do you have? How will you obtain them? Where from? Etc.
5. How will you structure your analysis in R?

Research Note

- Time is short so I don't expect a full-fledged paper but I do expect the following:
  - Be creative — explore a topic that you find interesting
  - Be transparent about your analysis. Use the functionality of R markdown to combine code and text
- If you don't know where to find data you can always use the EUSpeech data (for example the UK pm corpus)
- Devote a final section to specifically discuss strengths and weaknesses