

## Putting it all together on larger corpus

### Background

So far, we have been comparing two texts at the time. This simplifies learning NLP methodologies, but it's not a realistic scenario. For this assignment, I am going to give you **two options**:

1. **Freedom:** you can select your own small corpus (of around 20 texts) and design your own data exploration using the techniques we have learned in the class so far. That's the prompt: as long as you use the R techniques from the class, you get to pick your question, your texts, and your visualizations!
  - a. Who is this for? You took this class because you already have a project you are interested in or a specific question in mind or feel *very* comfortable with what we have been doing.
  - b. How will I be graded? Freedom comes with responsibility! Is your analysis and methodology appropriate for the questions you are asking? For your corpus? Are your analytic narrative and visualizations clear and to the point? These will be my criteria for grading. In addition, you can look at the structured prompt as a guideline for your assignment.
  - c. An extra requirement for this option: you may end up choosing a corpus that I and the TA's know nothing about! This means that you will have to keep that in mind in your write up. Make sure to give us enough context so that we can understand your analysis.
  - d. **Helpful hint:** run the idea for your data exploration by me to make sure that you are not getting yourself in trouble.
2. **Structured:** I will get into the details below (in the same vein as the homework assignment prompts). But the general idea is that you are going to start working with a slightly larger corpus: all the texts that I have uploaded on Canvas. The goal will be to bring together the techniques you learned so far in the class and explore how the texts in this corpus are (or are not) related to each other. **Important conceptual point:** we have reasons to think that these texts are related to each other because I selected them on purpose. They all are, in one way or another, “political economies” and we expect them to be in conversation with each other either explicitly or implicitly (they are all part of the same intellectual tradition).

### How to submit both versions:

Post your code on your GitHub. On **Canvas**: post your report (answering the interpretive questions along the way) in the Discussion section (Data Exploration). Make sure to include a link to the code **AND** upload **your visualizations/tables/etc.**

### Your Task:

---

**Purpose:** In this assignment, you will conduct a systematic exploratory analysis of our small corpus of texts using three complementary quantitative approaches:

1. TF-IDF (lexical distinctiveness)
2. Pearson correlation as a similarity / distance measure between texts
3. Syntactic complexity measures (Week 05 framework)

The idea behind this approach is to understand the corpus well enough to start generating good research questions. This assignment is designed to mirror how exploratory text analysis is actually used in humanities and social-science research: as a way to map structure, generate questions, and test intuitions, rather than to produce definitive answers.

### We will break this down into steps:

I will give hints where needed, but I will assume that you know what you are now much more comfortable with handling texts in tidy format.

#### Step 0—What to do about spelling and typographical issues?

I am going to give you a choice: in Week 3, the tutorial teaches you how to normalize Early Modern typography using regex. You are **required to** fix the long S in these texts! However, you have a choice as to whether you want to normalize other issues. Beyond this required step, you have a choice as to whether you want to normalize additional spelling or typographical variation (e.g. u/v, i/j, variant spellings, punctuation). **Note:** all these normalization choices carry some risks.

- Normalize the long S character.
- Carry out and explain your other normalization choices. Or explain if you choose not to change anything else. [There are good reasons for either option!]

#### Approach 1 — TF-IDF: lexical distinctiveness

##### Analytical role:

TF-IDF helps identify what makes each document distinctive relative to the corpus as a whole.

Required steps:

1. Construct a document–feature matrix (DFM) for the full corpus.
2. Compute TF-IDF weights.
3. For **each document**, extract the **top 10–15 TF-IDF terms**.

Required output:

- A table (or set of tables) showing distinctive terms by document.

Interpretive questions (address at least two):

- Do some documents share distinctive vocabulary?
- Are distinctive terms topical, rhetorical, or technical?

- Are there documents whose “distinctiveness” seems driven by noise or formatting rather than content?

Important: TF-IDF is **not** a measure of importance in the abstract. Interpret it relationally, always in reference to the rest of the corpus.

### Approach 2 — Pearson correlation: similarity and distance between texts

#### Analytical role:

As we discussed in class, Pearson correlation treats each document as a vector and asks: which texts behave similarly across the vocabulary space? We are going to focus on Pearson for this assignment since there were more questions about it in class!

Required steps:

1. Use the DFM to compute **pairwise Pearson correlations** between documents.
  - Optional: consider trimming very rare words from the DFM (e.g. `dfm_trim(min_termfreq = 5)`) before computing correlations. Very rare words helps ensure that Pearson correlations reflect shared patterns of language use, rather than being distorted by words that appear a few times due to non-Latin alphabet use (Greek) or idiosyncratic typesetting—this is due to the specific problems with the Early Modern texts we are using.
2. Visualize the results using similarity heatmap. [We didn’t cover this in the tutorials, but you can find a very generous hint at the end of this assignment!]

Required interpretation:

- Identify:
  - two most similar document pairs
  - two least similar document pairs
- What questions would you ask from this corpus after seeing the patterns of similarity/differences that you see here? Explain your answer.

Remember: in this use of Pearson correlations, you are evaluating patterns of similarity, not causation, as we discussed in class.

### Approach 3 — Syntactic complexity profile (Week 05)

#### Analytical role:

Syntactic complexity captures how texts are written, rather than what words they use. Note: I am only using the **syntactic complexity** measures as a way to make this more manageable. Because the

dependency parser (udpipe in our case) can take a long time to run depending on your machine (and silently crash!), we are going to focus on only two texts:

- pick two texts based on the results from the first two approaches (you have a lot of freedom to interpret what this means).

You must implement the **syntactic** complexity framework from Week 05, including:

- sentence length
- clause counts
- dependent clauses
- coordination
- complex nominals

Required measures:

- Mean Length of Sentence (MLS)
- Clauses per Sentence (C/S)
- Dependent Clauses per Clause and/or Sentence
- Coordination per Clause and/or Sentence
- Complex Nominals per Clause and/or Sentence

Required outputs:

- A summary table reporting all syntactic measures for both texts
- At least one example sentence from each text that illustrates a key syntactic difference you discuss

## Interpretation

In your report, address the following:

- How do the two texts differ in syntactic complexity?
- Do these differences align with or complicate your earlier lexical findings?
- What kinds of rhetorical or stylistic practices might these syntactic patterns reflect?

## Synthesis — triangulating evidence

---

This is the most important section of the report and it should bring everything together. You must articulate one central analytical question/hypothesis/idea that draws on:

- TF-IDF evidence
- Pearson similarity/distance evidence
- Syntactic complexity evidence

You have a lot of freedom in how to do this. I am not grading on your getting the “right” answer, but on demonstrating that you have carefully thought through the strengths and weaknesses of the methodologies as we discussed them in class.

---

### Coding hints for Approach 2: Pearson correlation + similarity heatmap (R)

---

#### What you are doing conceptually

You already built a **DFM** (named “`dfm_counts`”) where:

- rows = documents
- columns = features (words)
- cells = counts (or weighted count)

You will see a couple of new steps here

```
library(quanteda)
library(quanteda.textstats)
library(tidyverse)

# pairwise Pearson correlations between all documents
sim_r <- textstat_simil(dfm_counts, margin = "documents", method = "correlation")

# Convert similarity object to a matrix
r_mat <- as.matrix(sim_r)

# Take a quick look
dim(r_mat)
r_mat[1:5, 1:5]

# rounding correlation for readability (see explanation below)
```

```
r_mat <- round(r_mat, 3)

# Long format for ggplot

heat_df <- as.data.frame(r_mat) %>%
  rownames_to_column("doc_i") %>% # First document in pair
  pivot_longer(-doc_i, names_to = "doc_j", values_to = "r") # Second document +
  correlation

# create the heatmap

ggplot(heat_df, aes(x = doc_j, y = doc_i, fill = r)) +
  geom_tile() + # Create colored tiles
  coord_fixed() + # Keep tiles square
  scale_fill_gradient2(
    low = "blue",
    mid = "white",
    high = "red",
    midpoint = 0 # Center color scale at 0
  ) +
  labs( title = "Pearson Correlation Between Documents",
        x = NULL,
        y = NULL,
        fill = "Correlation"
  ) + theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    panel.grid = element_blank()
  )
```

How to read it:

- Darker / more intense = stronger similarity (higher  $r$ )
- The diagonal is always 1 (same document compared to itself)
- Blocks along the diagonal suggest clusters
- Tip: If document titles are long, shorten them before plotting (e.g., create a mapping to short labels).

**Why the rounding step in the code above:** Correlation coefficients like 0.56847392 are hard to read in tables and heatmaps. Rounding to 3 decimal places (0.568) keeps enough precision for interpretation while making patterns easier to spot visually. This is purely cosmetic and it doesn't change your analysis.