# Digital Humanities: Text-as-Data

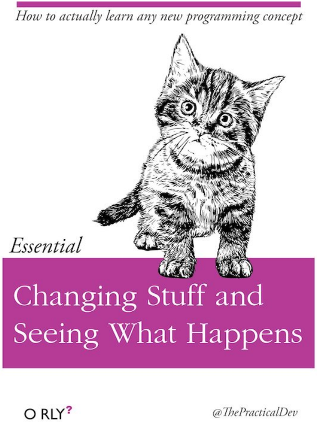
## Week 3 – Descriptive Patterns: Text Analysis and Hierarchical Clustering

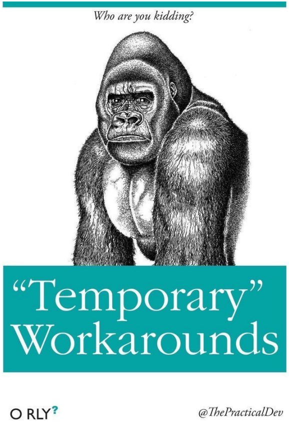
Steven Denney / Aron van de Pol
Leiden University

BA3 Korean Studies
October 24, 2025

# ODM usage, programming be like...

Software can be chaotic, but we make it work
Expert
Trying Stuff Until it Works
O RLY?
The Practical Developer
@ThePracticalDev


How to actually learn any new programming concept
Essential
Changing Stuff and Seeing What Happens
O RLY?
@ThePracticalDev



"*Keep clicking until it works*!"

# REVIEW

# Review: Text Preprocessing

- **Transformation** – lowercase, remove HTML/URLs, normalize characters.
- **Tokenization** – split text into tokens for counting.
- **Lemmatization** – reduce words to dictionary form.
- **Filtering** – remove stopwords and uninformative tokens.
- **POS tagging** – retain nouns or key grammatical categories.
- **N-grams** – detect multi-word expressions (e.g., "North Korea").

# Korean Text Preprocessing Solution (Optional)

**Orange’s native tools have limitations for Korean:**

- No Korean POS tagging – can’t distinguish nouns from particles.
- No Korean lemmatization – "먹다", "먹었다", "먹습니다" as different words.
- Forces regex workarounds – tedious and error-prone.

**We have made you a custom .py (python) script that does for you the following:**

- **Auto-installs** *kiwipiepy* (Korean NLP library).
- ***POS tagging** – identifies nouns, verbs, adjectives, adverbs.
- **Filters grammatical noise** – removes 40-60% of tokens (particles, endings).
- ***Lemmatizes** – extracts root forms regardless of conjugation.
- **Cleans** – removes URLs, emails, special characters, numbers and high/low frequencies.

# Korean Preprocessing: Location & Usage

## Location & Usage

`/data`

Copy .py script into Orange's Python Script widget.

See annotated version for detailed guidance.

Not working? Or you hate it maybe? No problem. Revert to regexp work-around.

# 1. BASICS OF TEXT ANALYSIS

# Term Frequency (TF)

## What it is

How many times a word appears in a document. Can be a raw count, but often "normalized".

- Measures how central a term is within one document.
- Shows common vocabulary, but not necessarily importance.
- Example: 중학교 국사 3차 (document)
  - 민족: 337
  - 운동: 297
  - 문화: 272

# Bag of Words (BoW)

### What it is

A text transformation method that converts a corpus of documents into a document-term matrix of word counts. The result is a ”bag” of words, where each document is represented by the counts of the words it contains.

| **document** | 역사 | 독립 | 근대화 | 민족 |
|---|---|---|---|---|
| 제1장 | 8 | 0 | 3 | 4 |
| 제2장 | 12 | 5 | 0 | 9 |
| 제3장 | 4 | 11 | 2 | 6 |

**Interpretation:** Frequency-based snapshot of vocabulary across documents.

# Document Frequency (DF)

## What it is

The number of documents in which a word appears at least once.

- Indicates how widespread or specialized a term is.
- Example (w/ 51 documents):

| **word** | **DF** | **Interpretation** |
|---|---|---|
| 역사 | 50 | almost ubiquitous |
| 독립운동 | 12 | specific to specific documents |
| 삼국시대 | 4 | concentrated in even more specific documents |

# Inverse Document Frequency (IDF)

## Plain Explanation

Measures how **distinctive** a word is across the corpus. Words that appear in many documents get low scores; rare words get high scores.

**Formula:**

$$IDF = \log\left(\frac{\text{total documents}}{\text{documents containing the word}}\right)$$

| Word | DF (out of 51) | IDF |
|---|---|---|
| 역사 | 50 | 0.02 (not distinctive) |
| 근대화 | 18 | 0.45 |
| 갑오개혁 | 4 | 1.10 (highly distinctive) |

# TF–IDF

## Plain Explanation

Combines two ideas:

- TF → how often a word appears in a document.
- IDF → how rare that word is across all documents.

**Formula:**

$$\text{TF–IDF} = \text{TF} \times \text{IDF}$$

**Meaning:** A high TF–IDF score = a rare and (maybe) important word.

| **Word** | TF | IDF | TF–IDF |
|---|---|---|---|
| 근대화 | 14 | 0.45 | 6.3 |
| 민주주의 | 11 | 0.52 | 5.7 |
| 역사 | 18 | 0.02 | 0.36 |

# Bag of Words in Orange Data Mining

## How to Configure BoW in Orange

The **Bag of Words** widget has three key settings that control word count processing:

- **Term Frequency:** How to count words
  - *Count*: Weighted word counts (default)
  - *Binary*: 1 if present, 0 if absent
  - *Sublinear*: Log of count
- **Document Frequency:** Weighting scheme
  - (*None*): No weighting
  - *IDF*: Downweight common words across documents
- **Regularization:** Normalization method (for more sophisticated analysis)

**Recommended:** `Count + IDF` for standard TF-IDF normalization

# Conceptual Summary

| **Measure** | **Focus** | **Penalizes** | **Use** |
|---|---|---|---|
| Word count | Frequency | – | Descriptive stats |
| TF (unweighted) | Frequency | – | Descriptive stats |
| TF (normalized) | Term prominence | Length | Descriptive stats/analysis |
| DF | Spread across docs | – | Corpus filtering |
| IDF | Common terms | High DF | Weighting |
| TF–IDF | Frequency × rarity | Common terms | Additional analysis |
| BoW | Representation | Context | Additional analysis |

# 2. CLUSTERING

# From Counting to Clustering

- Once we quantify words, we can measure how similar documents are.
- Clustering = automatically grouping documents that “talk alike.”
- Focus today: **Hierarchical Clustering.**

# What Is Hierarchical Clustering?

### Plain Explanation

Groups documents based on shared vocabulary patterns. Think of it as building a “family tree” of documents by similarity.

- Documents within the same cluster → similar content.
- Documents between clusters → different topics.

# How It Works (Conceptually)

1. Represent each document numerically (TF–IDF vectors).
2. Measure similarity (e.g., cosine distance).
3. Merge the most similar documents step-by-step.

The output is a **dendrogram** — a visual hierarchy of relationships.

# Pitfalls and Caveats

- Clustering depends on preprocessing choices (tokens, POS filters).
- Distance metric affects structure (cosine preferred for text).
- Over-clustering can reflect stylistic noise, not substance.
- TF–IDF weighting often yields more meaningful clusters than raw counts.

# Interpretive Takeaway

## From Description to Discovery

You are moving from counting words to identifying patterns that reflect underlying thematic or temporal structure in historical texts.

- **Descriptive:** Which words are frequent or distinctive?
- **Analytical:** Which documents are similar or different?
- **Interpretive:** What do these groupings reveal about historical narratives?

# ASSIGNMENT