

# Descriptive Patterns

BA3 Text as Data · Session 3 · 2 October 2026

---

Steven Denney

Korean Studies, Leiden University

## Class objective

---

By the end of class, you can represent texts as counts and TF-IDF weights, and use simple visuals to describe the corpus.

## Attendance and review questions

---

Scan the QR code with your phone and take the survey for attendance and review.

No (smart) phone? Ask me for a piece of paper.



**Kiwi** Splits 국민이 into 국민 + 이. The script keeps the noun 국민.

**Stopwords** 시간 is in the stopwords file, so Preprocess Text removes it.

**Bag of words** Each text becomes a count of each word. Word order is lost.

**IDF** Words found in almost every document get less weight.

**R** `x <- 5` stores the value 5 in the variable x.

Today you count the nouns and weight them.

# Roadmap for today

---

| When  | What                                    | Minutes |
|-------|-----------------------------------------|---------|
| 09:15 | Opening survey and review               | 10      |
| 09:25 | From words to numbers                   | 45      |
| 10:10 | Orange demonstration, counts and TF-IDF | 35      |
| 10:45 | Checklist and next week                 | 15      |

# Screens away

No computers or phones

### Moon Jae-in, first-anniversary message

“국민이 문재인정부를 세웠다는 사실을 결코 잊지 않겠습니다.”

May 2018 · 100-speech course corpus

After Session 2

국민

문재인

정부

사실

How can the computer deal with 100 speeches made of preprocessed documents like these?

# Bag of words

## Bag of words · BoW

A way of representing documents by which words they contain and how often they occur, ignoring word order.

**Sentence 1** 국민이 문재인정부를 세웠다

**Sentence 2** 문재인정부가 국민을 세웠다

**Same bag**

국민   문재인   정부

**Kept** Which words, and how often.

**Lost** Word order and the particles, so who did what is gone.

## The document-term matrix (DTM)

Bag of words transforms the preprocessed documents into a *document-term matrix*: one row per document, one column per word, and a count in each cell.

| Speech                        | 국민       | 경제 | 경찰        | 마을        | 평화 | 정의 |
|-------------------------------|----------|----|-----------|-----------|----|----|
| Lee, Police Day, 2011         | 15       | 4  | <b>39</b> | 0         | 0  | 0  |
| Park, Saemaul leaders, 2016   | 0        | 0  | 0         | <b>42</b> | 0  | 0  |
| Moon, first anniversary, 2018 | <b>9</b> | 0  | 0         | 0         | 2  | 2  |

The full matrix for the small speech corpus has 100 rows and 4,364 columns.

## Most cells in a DTM are zero

---

**96%**

of the 436,400 cells are 0

100 speeches  $\times$  4,364 words

Each speech uses a small part of the vocabulary. A matrix with mostly zeros is **sparse**. That is normal!

# Term frequency and document frequency

---

## Term frequency · TF

---

How often a word occurs in **one** speech.

국민 occurs 9 times in Moon's anniversary message.

## Document frequency · DF

---

In how **many** speeches a word occurs.

국민 is in 83 of 100 speeches.  
경찰 is in 5.

TF is the value in one cell. DF counts the documents with a non-zero value in a whole column.

## High frequency, little information

| Word            | Count | DF |
|-----------------|-------|----|
| 국민 (the people) | 443   | 83 |
| 정부 (government) | 312   | 75 |
| 감사 (thanks)     | 174   | 84 |
| 노력 (effort)     | 139   | 65 |
| 경찰 (police)     | 92    | 5  |

Words like 국민 and 감사 are frequent and occur in most speeches. They show what presidents say in general and little about any one speech.

경찰 is less frequent but occurs in only five speeches. It says much more about those five.

# Weighting

---

## Weight

A value that replaces the raw count in each cell, so that some words count more and others less.

Raw counts treat every occurrence equally. Weighting changes how much each word contributes.

TF-IDF gives more weight to words that are distinctive to a document and less weight to words that appear in many documents.

## Inverse document frequency

$$\text{IDF} = \ln(N / \text{DF})$$

N = 100 speeches

| Word | DF | IDF    |
|------|----|--------|
| 국민   | 83 | 0.1863 |
| 평화   | 29 | 1.2379 |
| 경찰   | 5  | 2.9957 |
| 청산   | 1  | 4.6052 |

The fewer speeches a word appears in, the higher its IDF. A word in every speech gets 0.

## TF-IDF = TF × IDF

Moon, first-anniversary message, 57 nouns, 42 different words

| Word            | TF | IDF    | TF-IDF |
|-----------------|----|--------|--------|
| 세상 (world)      | 3  | 2.2073 | 6.62   |
| 정의 (justice)    | 2  | 3.2189 | 6.44   |
| 광장 (square)     | 1  | 4.6052 | 4.61   |
| 국민 (the people) | 9  | 0.1863 | 1.68   |

국민 is 1st in this speech by count and 29th of 42 by TF-IDF. One use of 광장 outweighs nine uses of 국민.

# High and low TF-IDF

---

## High TF-IDF

---

Frequent in this speech  
and uncommon in the others.

The word helps distinguish this  
speech.

## Low TF-IDF

---

Infrequent in this speech,  
or common across many speeches.

The word does little to distinguish  
this speech.

Counts show what is frequent. TF-IDF shows what is distinctive.

## Counts and TF-IDF, same 100 speeches

### Counts

지역 협력 지원  
발전 감사 국가 나라 위기  
사회 경제 국민 세계 기업  
생각 한국 정부 대한민국  
산업 성장 노력

### TF-IDF

베트남 중국  
인권 경제 산업 여성 코로나  
기업 마을 경찰 양국 인공  
한국 지능 백신 지역 방송  
금융 국회 위기

Top 20 words, sized by mean count (left) and mean TF-IDF (right) per speech.

## Let op! What produces a high TF-IDF score

---

**경찰** Ranks 1st by TF-IDF. Of its 92 uses, 72 occur in just two speeches: Lee's 2011 Police Day speech (39) and Moon's 2022 speech to new police officers (33).

**Top 20** For 9 of the top 20 TF-IDF words, more than half of all uses occur in just two speeches. This is true of none of the top 20 words by raw count.

A high TF-IDF can be driven by just one or two speeches. Check those speeches before interpreting the word.

## Let op! Kiwi splits

---

**인공 + 지능** Kiwi split all 53 uses of 인공지능. 46 are in Moon's 2019 speech at an AI conference.

**새 + 마을** Kiwi split 20 of the 35 uses of 새마을 in Park's 2016 Saemaul speech. Without them, 마을 falls out of the top 20.

Preprocessing choices from Session 2 show up in today's weights.

## Seven terms you must know

---

|                             |                                                                                                |
|-----------------------------|------------------------------------------------------------------------------------------------|
| <b>Bag of words</b>         | Represents each text by its words and their counts, ignoring word order.                       |
| <b>Document-term matrix</b> | A table with one row per document and one column per word.                                     |
| <b>TF</b>                   | How often a word occurs in one document.                                                       |
| <b>DF</b>                   | The number of documents in which a word occurs.                                                |
| <b>Weight</b>               | A value that replaces or adjusts the raw count.                                                |
| <b>IDF</b>                  | $\ln(N / DF)$ . Higher when a word appears in fewer documents.                                 |
| <b>TF-IDF</b>               | $TF \times IDF$ . High when a word is frequent in one document but uncommon across the others. |

# Laptops open

Open your Demo 2 workflow. Let's build out together.

## Bag of Words settings in Orange

| Setting            | Options                  | Today         |
|--------------------|--------------------------|---------------|
| Term Frequency     | Count, Binary, Sublinear | <b>Count</b>  |
| Document Frequency | (None), IDF, Smooth IDF  | <b>IDF</b>    |
| Regularization     | (None), L1, L2           | <b>(None)</b> |

Bag of Words with Count and IDF gives TF-IDF. The counts come from a Word Cloud connected straight to Preprocess Text. Write these settings in your README.

## Other options, not used today

---

**Binary** 1 if the word occurs in the speech, 0 if it does not.

**Sublinear**  $1 + \ln(\text{count})$ . Large counts grow more slowly.

**Smooth IDF**  $\ln(1 + N / \text{DF})$ . A word in every speech gets 0.69 instead of 0.

**L1, L2** Rescale each speech's row so long and short speeches can be compared.

You will see these in Orange. Today we use Count, then IDF, with no regularization.

## Reading the Word Cloud in Orange

---

**Counts** The Word Cloud after Preprocess Text shows each word's total count. 국민 shows 443.

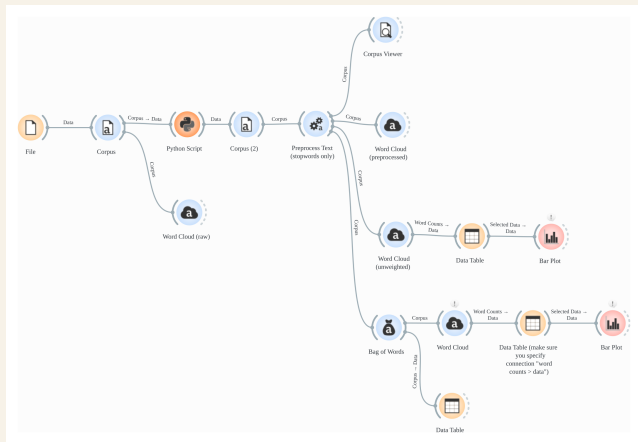
**TF-IDF** The Word Cloud after Bag of Words shows the mean weight per speech. 경찰 shows 2.76.

**Sizes** Only the top few words share the largest size. Use the table for ranks.

**Table** Link Word Counts to a Data Table. The weight is in the Word Count column, even for TF-IDF.

**Save** Save Image suggests .svg. Type the name with .png and click Save as Portable Network Graphics.

# Your Session 3 workflow in Orange



The snapshot is also on the Slides page.

**Task** The same two word clouds and two bar charts, made in R. Orange is not permitted.

**Sheet** The assignment sheet is on the Session 3 page, with every step and the code.

**New** `tidytext` adds the IDF. You multiply the count by it yourself.

Your numbers should be close to the ones on these slides.

# **Before you leave**

What must be finished, submitted, and prepared next

## Demo 3 checklist

---

- ☐ The workflow or script opens from the repository.
- ☐ The README names the corpus and the settings, and includes the reflection.
- ☐ `counts.png`, `tfidf.png`, `counts-top10.png`, and `tfidf-top10.png` are saved in `output/`.
- ☐ Everything is committed and pushed.

### **Demo 3 is due Oct. 9, 09:15**

Files go in `demos/03-descriptive-patterns/`. Graded for checklist completion.

## Next week compares speeches

---

### Demo 3

---

Finish the checklist and push.

Due Oct. 9 at 09:15.

Also watch the two Orange clustering tutorials. Details are on the Session 4 page.

### Programming exercises

---

DataCamp Introduction to R, chapters 2 and 3, checked online.

# **Appendix**

Key terms and definitions

## Appendix · Counting words

|                                   |                                                                                                 |
|-----------------------------------|-------------------------------------------------------------------------------------------------|
| <b>Bag of words (BoW)</b>         | A text represented by its words and their counts. Word order and grammar are dropped.           |
| <b>Document-term matrix (DTM)</b> | A table with one row per document and one column per word. Each cell holds a count or a weight. |
| <b>Sparse</b>                     | A matrix in which most cells are 0.                                                             |
| <b>Term frequency (TF)</b>        | How often a word occurs in one document.                                                        |
| <b>Document frequency (DF)</b>    | The number of documents in which a word occurs.                                                 |

## Appendix · Weighting words

### **Weight**

A number that replaces the raw count, so that some words count more and others less.

### **IDF**

$\ln(N / DF)$ , with  $N$  the number of documents. Rare words get a high IDF.

### **TF-IDF**

$TF \times IDF$ . A weight for words that are frequent in one document and rare in the others.

## Appendix · Orange widgets and settings

|                           |                                                                                                      |
|---------------------------|------------------------------------------------------------------------------------------------------|
| <b>Bag of Words</b>       | Builds the document-term matrix from a preprocessed corpus.                                          |
| <b>Term Frequency</b>     | Count uses the raw count. Binary uses 1 if the word occurs. Sublinear uses $1 + \ln(\text{count})$ . |
| <b>Document Frequency</b> | (None) keeps the term frequency. IDF multiplies it by $\ln(N / DF)$ .                                |
| <b>Regularization</b>     | Rescales each document's row. We use (None).                                                         |
| <b>Word Cloud</b>         | Sizes words by count, or by the mean Bag of Words weight when it follows Bag of Words.               |
| <b>Corpus Viewer</b>      | Finds the documents that contain a word.                                                             |