

Chapter 6 Working with Text Data

- Bag of words ignore word order
 - works well in moderate size texts,
not so much on long texts for overlap in words
- Stopwords: 'the, a, is, to, that'... Usually a set of most frequently occurring words in a corpus of text.
- Lemma: Canonical representation of a word, such as 'chase' of 'chased, chasing...'
 - ↳ Lemmatization.
- Stemming: Applying a set of deletion and substitution rules to that word to identify the word stem
 - ↳ Most consist of removing letters from end, obtaining 'chas' out of 'chases, chased'.
 - ↳ Porter Stemmer: Most popular stemmer
 - ↳ Removes the terminal 's'. Double 's' remain the same, such as 'chess'
 - ↳ Does not always find lemma - 'go' and 'went' can't be distinguished.
 - ↳ Replaces 'ies' at the end with 'i'. $\otimes$ parties $\rightarrow$ parti
- Higher order N-grams are usually for partially recovering word order.
- **Normalization** changes words or tokens to a canonical version so the words with only superficial differences are encoded the same way
 - ↳ $\otimes$ To lowercase

Rescaling Counts Using TF-IDF

- T: Term, D: Document. F: Frequency $\hookrightarrow$ useful for long texts
- $$TF = \begin{cases} 1 + \log(f_{t,d}) & \text{if } f_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$$

This is to dampen the effect of additional words.

$f_{t,d}$ = number of times t appears in document d .

If word appeared only once $\Rightarrow (1 + \log(1)) = 1$, if 100 times, $(1 + \log(100)) \approx 5.6$
- Rare words are more useful than frequently used words $\Rightarrow$ Inverse document frequency (IDF)
$$IDF = \log\left(\frac{N}{N_t}\right)$$
, where N = sample size and N_t = number of documents containing term t at least once. This function decreases as a function of term frequency

- $TF \cdot IDF$

↳ IDF: how many documents in the set contain the word.

$$(1 + \log(f_{t,d})) \times (\log \frac{N}{N_t})$$

Similarity Between Text Documents

- $\cos(90^\circ) = 0$ $\cos(0^\circ) = 1$
0: Documents are not similar. 1: Documents are identical

- $\cos(\theta) = \frac{\vec{x}_1 \cdot \vec{x}_2}{|\vec{x}_1| |\vec{x}_2|}$, $\theta = \text{angle}$, $x_1, x_2 = \text{Vector representation of two documents}$

⊗ $(2, 1, 1, 1, 0, 0, 0)$ and $(2, 0, 0, 0, 1, 1, 1)$

$$\vec{x}_1 \cdot \vec{x}_2 = 4 + 0 + 0 + 0 + 0 + 0 + 0 = 4$$

$$\|\vec{x}_1\| = \sqrt{5}, \text{ so } \sqrt{2^2 + 1^2 + 1^2 + 1^2} = \sqrt{7} \text{ and } \cos(\theta) = \frac{4}{\sqrt{5} \cdot \sqrt{7}} = 0.57$$

Mean 57% of their vector directions align, 57% of words are the same

- Cosine similarity is not affected by the length of the document