Term Frequency Weighting

Recall : Binary term-document incidence matrix

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

- Each document is represented by a binary vector $\in \{0,1\}^{|V|}$

Term-document count matrices

- Consider the number of occurrences of a term in a document:
  - Each document is a count vector $\mathbb{N}^{|V|}$: a column below. $\mathbb{N}$: the natural number

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	157	73	0	0	0	0
Brutus	4	157	0	1	0	0
Caesar	232	227	0	2	1	1
Calpurnia	0	10	0	0	0	0
Cleopatra	57	0	0	0	0	0
mercy	2	0	3	5	5	1
worser	2	0	1	1	1	0

Bag of words model

: term-document count model → Bag of words model : 순서 정보 무시

- Vector representation doesn't consider the ordering of words in a document
- "John is quicker than Mary" **and** "Mary is quicker than john" **have the same vectors**
- This is called the **bag of words** model
- the **positional index** was able to distinguish these two documents

Term Frequency (TF)

- The **term frequency** $tf_{t,d}$ of term t in document d is defined as **the number of times that** t **occurs in** d
- We want to use tf when computing **query-document match scores**.

- Raw term frequency is not what we want → **query와 document의 관련성(유사도)를 알고싶은 것**:
  - A document with 10 occurrences of the therm is **more relevant** than a document with 1 occurrence of the term.
  - But **not 10 times** more relevant
- **Relevance does not increase proportionally with term frequency**

Log-Frequency Weighting

tf가 높으면 높을수록 score이 커지되, linear 관계는 되지 않도록 **log**를 사용.

- The log frequency weight of term t in d is

$$w_{t,d} = \begin{cases} 1 + log_{10} tf_{t,d}, & \text{if } tf_{t,d} > 0 \\ 0, & \text{otherwise} \end{cases}$$

- 0 → 0, 1→1, 2→1.3, 10→2, 1000→4, etc.
- **Score for a document-query pair** : sum over terms t in both q and d :

  Score = $\sum_{t \in q \cap d} (1 + \log tf_{t,d})$
- The score is 0, if none of the query terms is present in the document

정리

- query와 document의 유사성을 좀 더 잘 반영하기 위해 term-frequency를 사용
- But, term-frequency는 bag of words model로 순서 정보 무시 → positional index를 사용하자
- But, term-frequency의 비율만큼 relevance이 늘어나는 것은 아님 → log term-frequency를 사용하자

출처 : standford IR 강의 (https://www.youtube.com/watch?v=9UXM2NXVYY0&list=PLaZQkZp6WhWwoDuD6pQCmgVyDbUWl_ZUi&index=9)