TF-IDF

TF-IDF is a metric that is calculated per term and document, meaning it is calculated for every pair of (term, document) in the corpus. Note that a document could be only a sentence.

It is composed of two parts: TF (term frequency) and IDF (inverse document frequency).

TF (Term Frequency)

TF (term frequency) ($tf_{t,d}$): How many times a term *t* appears in document *d*. It is simply a count.

How to represent it? To soften it, instead of the raw count, we represent it by (1+LOG10 count)

So, if a term appears 100 times in a document, their term frequency represented in logarithmic style will be $\mathbf{tf}_{t,d}$ = 1+LOG10(100) = 3.

DF (Document Frequency)

Document frequency (d_{ft}): Still for a given term, how many documents in the system contain this term? In other words, among all documents in the system (corpus), how many contain this term?

Document frequency as a ratio (a.k.a normalized document frequency): <u>what portion</u> of the documents in the system contain this term? If our corpus has 60 documents and 15 of them contain a specific term, the document frequency ratio for that term will be 15 out of 60 (15/60 = 0.25).

Inverse document frequency: Just the inverse of document frequency ratio, i.e., (N/df_t), so 60/15 = 4.

How to represent it? To soften it, instead of the raw inverse ratio, we represent it by LOG10 (N/ df_t)

For the example above, IDF_t = LOG10 (4) = 0.602

TF-IDF

TF-IDF $_{t,d}$ for a tuple of term and document: Multiply the logarithmic term frequency and logarithmic inverse document frequency!

So, for our example, it will be 0.25 * 0.602 = 0.1505

TF-IDF vector (TF-IDF $_d$) for a document: Vector of TF-IDF values for all terms in the vocabulary for this document. So, what we calculated was for one term. Assume the vocabulary in our model has 10 terms. So, we will have a vector of 10 values, i.e.,

$$\text{TF-IDF}_d = [\text{TF-IDF}_{t1,d}\ , \text{TF-IDF}_{t2,d}\ , \text{TF-IDF}_{t2,d}\ , \ldots, \text{TF-IDF}_{t10,d}\ ]$$

So, every document becomes a vector!

Normalized TF-IDF vector for a document: Normalize the vector above.

Cosine similarity between two documents: Dot product of the two normalized vectors of two documents.

Binary $\rightarrow$ count $\rightarrow$ weight matrix

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
Anthony	5.25	3.18	0.0	0.0	0.0	0.35	
Brutus	1.21	6.10	0.0	1.0	0.0	0.0	
Caesar	8.59	2.54	0.0	1.51	0.25	0.0	
Calpurnia	0.0	1.54	0.0	0.0	0.0	0.0	
Cleopatra	2.85	0.0	0.0	0.0	0.0	0.0	
mercy	1.51	0.0	1.90	0.12	5.25	0.88	
worser	1.37	0.0	0.11	4.15	0.25	1.95	
...							

Each document is now represented as a real-valued vector of tf-idf weights $\in \mathbb{R}^{|V|}$.

TF-IDF vector of the document "Anthony and Cleopatra" composed of 7 TF-IDF values (calculated for every term t and this doc)

All terms (vocabulary) of the model