TF-IDF for Machine Learning

Mustafa Coşkun

TF-IDF: term frequency-inverse document frequency

- TF-IDF stands for term frequency-inverse document frequency and it is a measure, used in the fields of information retrieval (IR) and machine learning, that can quantify the importance or relevance of string representations (words, phrases, lemmas, etc) in a document amongst a collection of documents

What is TF (term frequency)?

- Term frequency works by looking at the frequency of a particular term you are concerned with relative to the document. There are multiple measures, or ways, of defining frequency:
- Number of times the word appears in a document (raw count).
- Term frequency adjusted for the length of the document (raw count of occurences divided by number of words in the document).

TF: Term Frequency

Sent1: “good boy”

Sent2: “good girl”

Sent3: “boy girl good”

TF			
	Sent1	Sent2	Sent3
good	1/2	1/2	1/3
boy	1/2	0	1/3
girl	0	1/2	1/3

TF= NumOfRepWordsInSentence / NumOfWordsInSent

IDF: inverse document frequency

Sent1: “good boy”

Sent2: “good girl”

Sent3: “boy girl good”

Boy is important in sent1 than others semantically

IDF= log(NumOfSentences / NumOfSentencesContainingWords

TF			
	Sent1	Sent2	Sent3
good	1/2	1/2	1/3
boy	1/2	0	1/3
girl	0	1/2	1/3

*

IDF	
	IDF
good	log(3/3)=0
boy	log(3/2)=0.176
girl	log(3/2)=0.176

=

TF			
	Good	Boy	girl
sent1	0	0.08	0
sent2	0	0	0.08
sent3	0	0.06	0.06

Pros and cons of using TF-IDF

Pros of using TF-IDF

- The biggest advantages of TF-IDF come from how simple and easy to use it is. It is simple to calculate, it is computationally cheap, and it is a simple starting point for similarity calculations (via TF-IDF vectorization + cosine similarity).

Cons of using TF-IDF

- Something to be aware of is that TF-IDF cannot help carry semantic meaning. It considers the importance of the words due to how it weighs them, but it cannot necessarily derive the contexts of the words and understand importance that way.
- Also, TF-IDF ignores word order and thus compound nouns like “Queen of England” will not be considered as a “single unit”. This also extends to situations like negation with “not pay the bill” vs “pay the bill”, where the order makes a big difference. In both cases using underscores, “queen_of_england” or “not_pay” are ways to handle treating the phrase as a single unit.
- Another disadvantage is that it can suffer from memory-inefficiency since TF-IDF can suffer from the curse of dimensionality. Recall that the length of TF-IDF vectors is equal to the size of the vocabulary. In some classification contexts this may not be an issue but in other contexts like clustering this can be unwieldy as the number of documents increases. Thus looking into some of the alternatives (BERT, Word2Vec) may be necessary.

https://www.capitalone.com/tech/machine-learning/understanding-tf-idf/