tf–idf

From Wikipedia, the free encyclopedia
(Redirected from Tfidf)

The **tf–idf** weight (term frequency–inverse document frequency) is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the tf–idf weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. tf–idf can be successfully used for stop-words filtering in various subject fields including text summarization and classification[1].

One of the simplest ranking functions is computed by summing the tf–idf for each query term; many more sophisticated ranking functions are variants of this simple model.

Contents

Motivation

Suppose we have a set of English text documents and wish to determine which document is most relevant to the query "the brown cow." A simple way to start out is by eliminating documents that do not contain all three words "the," "brown," and "cow," but this still leaves many documents. To further distinguish them, we might count the number of times each term occurs in each document and sum them all together; the number of times a term occurs in a document is called its *term frequency*. However, because the term "the" is so common, this will tend to incorrectly emphasize documents which happen to use the word "the" more frequently, without giving enough weight to the more meaningful terms "brown" and "cow". Also the term "the" is not a good keyword to distinguish relevant and non-relevant documents and terms. On the contrary, the words "brown" and "cow" that occur rarely are good keywords to distinguish relevant documents from the non-relevant documents. Hence an *inverse document frequency* factor is incorporated which diminishes the weight of terms that occur very frequently in the collection and increases the weight of terms that occur rarely.

Mathematical details

The *term count* in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents (which may have a higher term count regardless of the actual importance of that term in the document) to give a measure of the importance of the term t within the particular document d. Thus we have the *term frequency* $\mathrm{tf}(t,d)$, defined in the simplest case as the occurrence count of a term in a document. (Many variants have been suggested; see e.g. Manning, Raghavan and

Schütze, p. 118. (http://nlp.stanford.edu/IR-book/html/htmledition/document-and-query-weighting-schemes-1.html))

The *inverse document frequency* is a measure of the general importance of the term (obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient).

$$\mathrm{idf}(t) = \log \frac{|D|}{|\{d : t \in d\}|}$$

with

- $|D|$: cardinality of D, or the total number of documents in the corpus
- $|\{d : t \in d\}|$: number of documents where the term t appears (i.e., $\mathrm{tf}(t, d) \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the formula to $1 + |\{d : t \in d\}|$.

Then

$$\text{tf-idf}(t, d) = \mathrm{tf}(t, d) \times \mathrm{idf}(t)$$

A high weight in tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weights hence tend to filter out common terms. The tf–idf value for a term will be greater than zero if and only if the ratio inside the idf's log function is greater than 1. Depending on whether a 1 is added to the denominator, a term in all documents will have either a zero or negative idf, and if the 1 is added to the denominator a term that occurs in all but one document will have an idf equal to zero.

Various (mathematical) forms of the tf–idf term weight can be derived from a probabilistic retrieval model that mimicks human relevance decision making.

Example

Consider a document containing 100 words wherein the word *cow* appears 3 times. Following the previously defined formulas, the term frequency (TF) for *cow* is then (3 / 100) = 0.03. Now, assume we have 10 million documents and *cow* appears in one thousand of these. Then, the inverse document frequency is calculated as log(10 000 000 / 1 000) = 4. The tf–idf score is the product of these quantities: 0.03 × 4 = 0.12.

See also

- Okapi BM25
- Noun phrase
- Word count
- Kullback-Leibler divergence
- Mutual Information
- Latent semantic analysis
- Latent semantic indexing
- Latent Dirichlet allocation

References

- Spärck Jones, Karen (1972). "A statistical interpretation of term specificity and its application in retrieval" (http://www.soi.city.ac.uk/~ser/idfpapers/ksj_orig.pdf) . *Journal of Documentation* **28** (1): 11–21. doi:10.1108/eb026526 (http://dx.doi.org/10.1108%2Feb026526) . http://www.soi.city.ac.uk/~ser/idfpapers/ksj_orig.pdf.
- Salton, G. and M. J. McGill (1983). *Introduction to modern information retrieval*. McGraw-Hill. ISBN 0070544840.
- Salton, Gerard, Edward A. Fox & Harry Wu (November 1983). "Extended Boolean information retrieval" (http://portal.acm.org/citation.cfm?id=358466) . *Communications of the ACM* **26** (11): 1022–1036. doi:10.1145/182.358466 (http://dx.doi.org/10.1145%2F182.358466) . http://portal.acm.org/citation.cfm?id=358466.
- Salton, Gerard and Buckley, C. (1988). "Term-weighting approaches in automatic text retrieval". *Information Processing & Management* **24** (5): 513–523. doi:10.1016/0306-4573(88)90021-0 (http://dx.doi.org/10.1016%2F0306-4573%2888%2990021-0) .
- H.C. Wu, R.W.P. Luk, K.F. Wong, K.L. Kwok (2008). "Interpreting tf–idf term weights as making relevance decisions". *ACM Transactions on Information Systems* **26** (3): 1–37. doi:10.1145/1361684.1361686 (http://dx.doi.org/10.1145%2F1361684.1361686) .

External links

- Gensim (http://nlp.fi.muni.cz/projekty/gensim) is a Python+NumPy framework for Vector Space modelling. It contains incremental (memory-efficient) algorithms for Tf–idf, Latent Semantic Indexing and Latent Dirichlet Allocation.
- Term Weighting Approaches in Automatic Text Retrieval (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.101.9086)
- Robust Hyperlinking (http://bscit.berkeley.edu/cgi-bin/pl_dochome?query_src=&format=html&collection=Wilensky_papers&id=3&show_doc=yes) : An application of tf–idf for stable document addressability.
- A demo of using tf–idf with PHP and Euclidean distance for Classification (http://infinova.wordpress.com/2010/01/26/distance-between-documents/)
- Anatomy of a search engine (http://www.codeproject.com/KB/IP/AnatomyOfASearchEngine1.aspx)
- tf–idf and related definitions (http://lucene.apache.org/java/3_2_0/api/core/org/apache/lucene/search/Similarity.html) as used in Lucene
- tf–idf support in scikit-learn (http://scikit-learn.sourceforge.net/modules/generated/scikits.learn.feature_extraction.text.TfidfTransformer.html#scikits.
- Text to Matrix Generator (TMG) (http://scgroup.hpclab.ceid.upatras.gr/scgroup/Projects/TMG/) MATLAB toolbox that can be used for various tasks in text mining (TM) specifically i) indexing, ii) retrieval, iii) dimensionality reduction, iv) clustering, v) classification. The indexing step offers the user the ability to apply local and global weighting methods, including tf–idf.
- Pyevolve: A tutorial series explaining the tf-idf calculation (http://pyevolve.sourceforge.net/wordpress/?p=1589) .

1. ^ TF*IDF Ranker (http://vetsky.narod2.ru/catalog/tfidf_ranker/)

Retrieved from "http://en.wikipedia.org/w/index.php?title=Tf%E2%80%93idf&oldid=457500322"

Categories: Information retrieval | Artificial intelligence applications | Statistical natural language processing | Ranking functions

---

- This page was last modified on 26 October 2011 at 15:29.

- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. See Terms of use for details.
  Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.