WIKIPEDIA

tf–idf

In information retrieval, **tf–idf** or **TFIDF**, short for **term frequency–inverse document frequency**, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus.[1] It is often used as a weighting factor in searches of information retrieval, text mining, and user modeling. The tf–idf value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps to adjust for the fact that some words appear more frequently in general. tf–idf is one of the most popular term-weighting schemes today. A survey conducted in 2015 showed that 83% of text-based recommender systems in digital libraries use tf–idf.[2]

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.

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

Motivations
- Term frequency
- Inverse document frequency

Definition
- Term frequency
- Inverse document frequency
- Term frequency–Inverse document frequency

Justification of idf

Link with Information Theory

Example of tf–idf

Beyond terms

Derivatives

See also

References

External links and suggested reading

Motivations

Term frequency

Suppose we have a set of English text documents and wish to rank 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; the number of times a term occurs in a document is called its *term frequency*. However, in the case where the length of documents varies greatly, adjustments are often made (see definition below). The first form of term weighting is due to Hans Peter Luhn (1957) which may be summarized as:[3]

> The weight of a term that occurs in a document is simply proportional to the term frequency.

Inverse document frequency

Because the term "the" is so common, term frequency 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". The term "the" is not a good keyword to distinguish relevant and non-relevant documents and terms, unlike the less-common words "brown" and "cow". Hence an *inverse document frequency* factor is incorporated which diminishes the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely.

Karen Spärck Jones (1972) conceived a statistical interpretation of term-specificity called Inverse Document Frequency (idf), which became a cornerstone of term weighting:[4]

> The specificity of a term can be quantified as an inverse function of the number of documents in which it occurs.

Definition

1. The tf–idf is the product of two statistics, *term frequency* and *inverse document frequency*. There are various ways for determining the exact values of both statistics.
2. A formula that aims to define the importance of a keyword or phrase within a document or a web page.

Term frequency

In the case of the **term frequency** $\mathrm{tf}(t,d)$, the simplest choice is to use the *raw count* of a term in a document, i.e., the number of times that term t occurs in document d. If we denote the raw count by $f_{t,d}$, then the simplest tf scheme is $\mathrm{tf}(t,d) = f_{t,d}$. Other possibilities include[5]:128

- Boolean "frequencies": $\mathrm{tf}(t,d) = 1$ if t occurs in d and 0 otherwise;
- term frequency adjusted for document length : $f_{t,d}$ ÷ (number of words in d)
- logarithmically scaled frequency: $\mathrm{tf}(t,d) = \log(1 + f_{t,d})$;[6]
- augmented frequency, to prevent a bias towards longer documents, e.g. raw frequency divided by the raw frequency of the most occurring term in the document:

$$\mathrm{tf}(t,d) = 0.5 + 0.5 \cdot \frac{f_{t,d}}{\max\{f_{t',d} : t' \in d\}}$$

Variants of term frequency (tf) weight

weighting scheme	tf weight
binary	$0, 1$
raw count	$f_{t,d}$
term frequency	$f_{t,d} \Big/ \sum_{t' \in d} f_{t',d}$
log normalization	$\log(1 + f_{t,d})$
double normalization 0.5	$0.5 + 0.5 \cdot \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$
double normalization K	$K + (1 - K)\frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$

Inverse document frequency

The **inverse document frequency** is a measure of how much information the word provides, i.e., if it's common or rare across all documents. It is the logarithmically scaled inverse fraction of the documents that contain the word (obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient):

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

with

- N: total number of documents in the corpus $N = |D|$
- $|\{d \in D : t \in d\}|$: 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 denominator to $1 + |\{d \in D : t \in d\}|$.

Variants of inverse document frequency (idf) weight

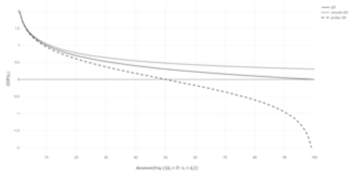
weighting scheme	idf weight ($n_t = \|\{d \in D : t \in d\}\|$)
unary	1
inverse document frequency	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$
inverse document frequency smooth	$\log\left(\frac{N}{1 + n_t}\right) + 1$
inverse document frequency max	$\log\left(\frac{\max_{\{t' \in d\}} n_{t'}}{1 + n_t}\right)$
probabilistic inverse document frequency	$\log \frac{N - n_t}{n_t}$

Term frequency–Inverse document frequency

Then tf–idf is calculated as

$$\mathrm{tfidf}(t,d,D) = \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,D)$$

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. Since the ratio inside the idf's log function is always greater than or equal to 1, the value of idf (and tf–idf) is greater than or equal to 0. As a term appears in more documents, the ratio inside the logarithm approaches 1, bringing the idf and tf–idf closer to 0.



Plot of different inverse document frequency functions: standard, smooth, probabilistic.

Recommended tf–idf weighting schemes

weighting scheme	document term weight	query term weight
1	$f_{t,d} \cdot \log \frac{N}{n_t}$	$\left(0.5 + 0.5\frac{f_{t,q}}{\max_t f_{t,q}}\right) \cdot \log \frac{N}{n_t}$
2	$1 + \log f_{t,d}$	$\log\left(1 + \frac{N}{n_t}\right)$
3	$(1 + \log f_{t,d}) \cdot \log \frac{N}{n_t}$	$(1 + \log f_{t,q}) \cdot \log \frac{N}{n_t}$

Justification of idf

Idf was introduced, as "term specificity", by Karen Spärck Jones in a 1972 paper. Although it has worked well as a heuristic, its theoretical foundations have been troublesome for at least three decades afterward, with many researchers trying to find information theoretic justifications for it.[7]

Spärck Jones's own explanation did not propose much theory, aside from a connection to Zipf's law.[7] Attempts have been made to put idf on a probabilistic footing,[8] by estimating the probability that a given document d contains a term t as the relative document frequency,

$$P(t|D) = \frac{|\{d \in D : t \in d\}|}{N},$$

so that we can define idf as

$$\begin{aligned}\mathrm{idf} &= -\log P(t|D) \\ &= \log \frac{1}{P(t|D)} \\ &= \log \frac{N}{|\{d \in D : t \in d\}|}\end{aligned}$$

Namely, the inverse document frequency is the logarithm of "inverse" relative document frequency.

This probabilistic interpretation in turn takes the same form as that of self-information. However, applying such information-theoretic notions to problems in information retrieval leads to problems when trying to define the appropriate event spaces for the required probability distributions: not only documents need to be taken into account, but also queries and terms.[7]

Link with Information Theory

The Term Frequency and the Inverse Document Frequency can be formulated using Information theory; it helps to understand why their product have a meaning in terms of joint informational content of a document. A characteristic assumption about the distribution $p(d,t)$ is that:

$$p(d|t) = \frac{1}{|\{d \in D : t \in d\}|}$$

This assumption and its implications, according to Aizawa: "represent the heuristic that tf-idf employs."[9]

Recall the expression of the Conditional entropy of a "randomly chosen" document in the corpus D conditional to the fact it contains a specific term t (and assume that all documents have equal probability to be chosen, and small p being r=probabilities)):

$$H(\mathcal{D}|\mathcal{T}=t) = -\sum_d p_{d|t} \log p_{d|t} = -\log \frac{1}{|\{d \in D : t \in d\}|} = \log \frac{|\{d \in D : t \in d\}|}{|D|} + \log |D| = -\mathrm{idf}(t) + \log |D|$$

In terms of notation, $\mathcal{D}$ and $\mathcal{T}$ are "random variables" corresponding to respectively draw a document or a term. Now recall the definition of the Mutual information and note that it can be expressed as

$$M(\mathcal{T};\mathcal{D}) = H(\mathcal{D}) - H(\mathcal{D}|\mathcal{T}) = \sum_t p_t \cdot (H(\mathcal{D}) - H(\mathcal{D}|W=t)) = \sum_t p_t \cdot \mathrm{idf}(t)$$

The last step is to expand p_t, the unconditional probability to draw a term, with respect to the (random) choice of a document, to obtain:

$$M(\mathcal{T};\mathcal{D}) = \sum_{t,d} p_{t|d} \cdot p_d \cdot \mathrm{idf}(t) = \sum_{t,d} \mathrm{tf}(t,d) \cdot \frac{1}{|D|} \cdot \mathrm{idf}(t) = \frac{1}{|D|} \sum_{t,d} \mathrm{tf}(t,d) \cdot \mathrm{idf}(t).$$

This expression shows that summing the Tf-idf of all possible terms and documents recovers the mutual information between documents and term taking into account all the specificities of their joint distribution (for details, see.[10] Each Tf-idf hence carries the "bit of information" attached to a term x document pair.

Example of tf–idf

Suppose that we have term count tables of a corpus consisting of only two documents, as listed on the right.

The calculation of tf–idf for the term "this" is performed as follows:

In its raw frequency form, tf is just the frequency of the "this" for each document. In each document, the word "this" appears once; but as the document 2 has more words, its relative frequency is smaller.

Document 1

Term	Term Count
this	1
is	1
a	2
sample	1

Document 2

Term	Term Count
this	1
is	1
another	2
example	3

$$\mathrm{tf}(''\textsf{this}'', d_1) = \frac{1}{5} = 0.2$$
$$\mathrm{tf}(''\textsf{this}'', d_2) = \frac{1}{7} \approx 0.14$$

An idf is constant per corpus, and **accounts** for the ratio of documents that include the word "this". In this case, we have a corpus of two documents and all of them include the word "this".

$$\mathrm{idf}(''\textsf{this}'', D) = \log\left(\frac{2}{2}\right) = 0$$

So tf–idf is zero for the word "this", which implies that the word is not very informative as it appears in all documents.

$$\mathrm{tfidf}(''\textsf{this}'', d_1, D) = 0.2 \times 0 = 0$$
$$\mathrm{tfidf}(''\textsf{this}'', d_2, D) = 0.14 \times 0 = 0$$

The word "example" is more interesting - it occurs three times, but only in the second document:

$$\mathrm{tf}(''\textsf{example}'', d_1) = \frac{0}{5} = 0$$
$$\mathrm{tf}(''\textsf{example}'', d_2) = \frac{3}{7} \approx 0.429$$
$$\mathrm{idf}(''\textsf{example}'', D) = \log\left(\frac{2}{1}\right) = 0.301$$

Finally,

$$\mathrm{tfidf}(''\textsf{example}'', d_1, D) = \mathrm{tf}(''\textsf{example}'', d_1) \times \mathrm{idf}(''\textsf{example}'', D) = 0 \times 0.301 = 0$$
$$\mathrm{tfidf}(''\textsf{example}'', d_2, D) = \mathrm{tf}(''\textsf{example}'', d_2) \times \mathrm{idf}(''\textsf{example}'', D) = 0.429 \times 0.301 \approx 0.129$$

(using the base 10 logarithm).

Beyond terms

The idea behind tf–idf also applies to entities other than terms. In 1998, the concept of idf was applied to citations.[11] The authors argued that "if a very uncommon citation is shared by two documents, this should be weighted more highly than a citation made by a large number of documents". In addition, tf–idf was applied to "visual words" with the purpose of conducting object matching in videos,[12] and entire sentences.[13] However, the concept of tf–idf did not prove to be more effective in all cases than a plain tf scheme (without idf). When tf–idf was applied to citations, researchers could find no improvement over a simple citation-count weight that had no idf component.[14]

Derivatives

A number of term-weighting schemes have derived from tf–idf. One of them is TF–PDF (Term Frequency * Proportional Document Frequency).[15] TF–PDF was introduced in 2001 in the context of identifying emerging topics in the media. The PDF component measures the difference of how often a term occurs in different domains. Another derivate is TF–IDuF. In TF–IDuF,[16] idf is not calculated based on the document corpus that is to be searched or recommended. Instead, idf is calculated on users' personal document collections. The authors report that TF–IDuF was equally effective as tf–idf but could also be applied in situations when, e.g., a user modeling system has no access to a global document corpus.

See also

- Word embedding
- Kullback–Leibler divergence
- Latent Dirichlet allocation
- Latent semantic analysis
- Mutual information
- Noun phrase
- Okapi BM25
- PageRank
- Vector space model
- Word count
- SMART Information Retrieval System

References

1. Rajaraman, A.; Ullman, J.D. (2011). "Data Mining" (http://i.stanford.edu/~ullman/mmds/ch1.pdf) (PDF). *Mining of Massive Datasets*. pp. 1–17. doi:10.1017/CBO9781139058452.002 (https://doi.org/10.1017%2FCBO9781139058452.002). ISBN 978-1-139-05845-2.
2. Breitinger, Corinna; Gipp, Bela; Langer, Stefan (2015-07-26). "Research-paper recommender systems: a literature survey" (http://nbn-resolving.de/urn:nbn:de:bsz:352-0-311312). *International Journal on Digital Libraries*. **17** (4): 305–338. doi:10.1007/s00799-015-0156-0 (https://doi.org/10.1007%2Fs00799-015-0156-0). ISSN 1432-5012 (https://www.worldcat.org/issn/1432-5012). S2CID 207035184 (https://api.semanticscholar.org/CorpusID:207035184).
3. Luhn, Hans Peter (1957). "A Statistical Approach to Mechanized Encoding and Searching of Literary Information" (http://web.stanford.edu/class/linguist289/luhn57.pdf) (PDF). *IBM Journal of Research and Development*. **1** (4): 309–317. doi:10.1147/rd.14.0309 (https://doi.org/10.1147%2Frd.14.0309). Retrieved 2 March 2015. "There is also the probability that the more frequently a notion and combination of notions occur, the more importance the author attaches to them as reflecting the essence of his overall idea."
4. Spärck Jones, K. (1972). "A Statistical Interpretation of Term Specificity and Its Application in Retrieval". *Journal of Documentation*. **28**: 11–21. CiteSeerX 10.1.1.115.8343 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.115.8343). doi:10.1108/eb026526 (https://doi.org/10.1108%2Feb026526).
5. Manning, C.D.; Raghavan, P.; Schutze, H. (2008). "Scoring, term weighting, and the vector space model" (http://nlp.stanford.edu/IR-book/pdf/06vect.pdf) (PDF). *Introduction to Information Retrieval*. p. 100. doi:10.1017/CBO9780511809071.007 (https://doi.org/10.1017%2FCBO9780511809071.007). ISBN 978-0-511-80907-1.
6. "TFIDF statistics | SAX-VSM" (https://jmotif.github.io/sax-vsm_site/morea/algorithm/TFIDF.html).
7. Robertson, S. (2004). "Understanding inverse document frequency: On theoretical arguments for IDF". *Journal of Documentation*. **60** (5): 503–520. doi:10.1108/00220410410560582 (https://doi.org/10.1108%2F00220410410560582).
8. See also Probability estimates in practice (http://nlp.stanford.edu/IR-book/html/htmledition/probability-estimates-in-practice-1.html#p:justificationofidf) in *Introduction to Information Retrieval*.
9. Aizawa, Akiko (2003). "An information-theoretic perspective of tf–idf measures". *Information Processing and Management*. **39** (1): 45–65. doi:10.1016/S0306-4573(02)00021-3 (https://doi.org/10.1016%2FS0306-4573%2802%2900021-3).
10. Aizawa, Akiko (2003). "An information-theoretic perspective of tf–idf measures". *Information Processing and Management*. **39** (1): 45–65. doi:10.1016/S0306-4573(02)00021-3 (https://doi.org/10.1016%2FS0306-4573%2802%2900021-3).
11. Bollacker, Kurt D.; Lawrence, Steve; Giles, C. Lee (1998-01-01). *CiteSeer: An Autonomous Web Agent for Automatic Retrieval and Identification of Interesting Publications* (https://www.semanticscholar.org/paper/b23a5a62b7cb5278ceb5a6cc021c28a92041d792). *Proceedings of the Second International Conference on Autonomous Agents*. AGENTS '98. pp. 116–123. doi:10.1145/280765.280786 (https://doi.org/10.1145%2F280765.280786). ISBN 978-0-89791-983-8. S2CID 3526393 (https://api.semanticscholar.org/CorpusID:3526393).
12. Sivic, Josef; Zisserman, Andrew (2003-01-01). *Video Google: A Text Retrieval Approach to Object Matching in Videos* (http://dl.acm.org/citation.cfm?id=946247.946751). *Proceedings of the Ninth IEEE International Conference on Computer Vision – Volume 2*. ICCV '03. pp. 1470–. doi:10.1109/ICCV.2003.1238663 (https://doi.org/10.1109%2FICCV.2003.1238663). ISBN 978-0-7695-1950-0. S2CID 14457153 (https://api.semanticscholar.org/CorpusID:14457153).
13. Seki, Yohei. "Sentence Extraction by tf/idf and Position Weighting from Newspaper Articles" (http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings3/NTCIR3-TSC-SekiY.pdf) (PDF). National Institute of Informatics.
14. Beel, Joeran; Breitinger, Corinna (2017). "Evaluating the CC-IDF citation-weighting scheme – How effectively can 'Inverse Document Frequency' (IDF) be applied to references?" (http://beel.org/publications/2017%20iConference%20--%20Evaluating%20the%20CC-IDF%20citation-weighting%20scheme%20--%20preprint.pdf) (PDF). *Proceedings of the 12th IConference*.

15. Khoo Khyou Bun; Bun, Khoo Khyou; Ishizuka, M. (2001). *Emerging Topic Tracking System*. *Proceedings Third International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems. WECWIS 2001*. p. 2. CiteSeerX 10.1.1.16.7986 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.16.7986). doi:10.1109/wecwis.2001.933900 (https://doi.org/10.1109%2Fwecwis.2001.933900). ISBN 978-0-7695-1224-2. S2CID 1049263 (https://api.semanticscholar.org/CorpusID:1049263).
16. Langer, Stefan; Gipp, Bela (2017). "TF-IDuF: A Novel Term-Weighting Scheme for User Modeling based on Users' Personal Document Collections" (https://www.gipp.com/wp-content/papercite-data/pdf/beel17.pdf) (PDF). *IConference*.

- Salton, G; McGill, M. J. (1986). *Introduction to modern information retrieval* (https://archive.org/details/introductiontomo00salt). McGraw-Hill. ISBN 978-0-07-054484-0.
- Salton, G.; Fox, E. A.; Wu, H. (1983). "Extended Boolean information retrieval". *Communications of the ACM*. **26** (11): 1022–1036. doi:10.1145/182.358466 (https://doi.org/10.1145%2F182.358466). hdl:1813/6351 (https://hdl.handle.net/1813%2F6351). S2CID 207180535 (https://api.semanticscholar.org/CorpusID:207180535).
- Salton, G.; Buckley, C. (1988). "Term-weighting approaches in automatic text retrieval" (https://ecommons.cornell.edu/bitstream/1813/6721/1/87-881.pdf) (PDF). *Information Processing & Management*. **24** (5): 513–523. doi:10.1016/0306-4573(88)90021-0 (https://doi.org/10.1016%2F0306-4573%2888%2990021-0). hdl:1813/6721 (https://hdl.handle.net/1813%2F6721).
- Wu, H. C.; Luk, R.W.P.; Wong, K.F.; Kwok, K.L. (2008). "Interpreting TF-IDF term weights as making relevance decisions" (https://www.semanticscholar.org/paper/f6bbbf2cc785cf96019dcd9c41ab1801aad962dd). *ACM Transactions on Information Systems*. **26** (3): 1. doi:10.1145/1361684.1361686 (https://doi.org/10.1145%2F1361684.1361686). hdl:10397/10130 (https://hdl.handle.net/10397%2F10130). S2CID 18303048 (https://api.semanticscholar.org/CorpusID:18303048).

External links and suggested reading

- Gensim is a Python library for vector space modeling and includes tf–idf weighting.
- 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.
- Anatomy of a search engine (http://www.codeproject.com/KB/IP/AnatomyOfASearchEngine1.aspx)
- tf–idf and related definitions (http://lucene.apache.org/core/3_6_1/api/all/org/apache/lucene/search/Similarity.html) as used in Lucene
- TfidfTransformer (http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html#sklearn.feature_extraction.text.TfidfTransformer) in scikit-learn
- 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.