1. 词集模型(set of word): 文档单词构成的集合，词集中的每个单词如果出现就置1，没有就置0，每个单词只出现一次，（也就是没有词频的概念）。
2. 词袋模型(bag of word)：文档单词构成的集合，但统计词频，也就是说各个单词的权重不一样的，由词频决定
3. tf-idf：单词的权重是对应单词的tf-idf值

TF-IDF(Term Frequency-Inverse Document Frequency, 词频-逆文件频率).

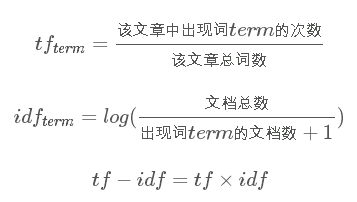
是一种用于资讯检索与资讯探勘的常用加权技术。TF-IDF是一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。

上述引用总结就是, **一个词语在一篇文章中出现次数越多, 同时在所有文档中出现次数越少, 越能够代表该文章.**

词频 (term frequency, TF) 指的是某一个给定的词语在该文件中出现的次数。这个数字通常会被归一化(一般是词频除以文章总词数), 以防止它偏向长的文件。（同一个词语在长文件里可能会比短文件有更高的词频，而不管该词语重要与否。）

但是, 需要注意, 一些通用的词语对于主题并没有太大的作用, 反倒是一些出现频率较少的词才能够表达文章的主题, 所以单纯使用是TF不合适的。权重的设计必须满足：一个词预测主题的能力越强，权重越大，反之，权重越小。所有统计的文章中，一些词只是在其中很少几篇文章中出现，那么这样的词对文章的主题的作用很大，这些词的权重应该设计的较大。IDF就是在完成这样的工作.

逆向文件频率 (inverse document frequency, IDF) IDF的主要思想是：如果包含词条t的文档越少, IDF越大，则说明词条具有很好的类别区分能力。某一特定词语的IDF，可以由总文件数目除以包含该词语之文件的数目，再将得到的商取对数得到。



IDF=log(语料库的文档总数包含词条w的文档数+1),分母之所以要加1，是为了避免分母为0

　　某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的TF-IDF。因此，TF-IDF倾向于过滤掉常见的词语，保留重要的词语。

1. 示例

假设从语料库中得到一个字典dict:

|  |  |
| --- | --- |
| 1 | [I, CS, like, NLP, and] |

现在有语料库中有两个文档

|  |  |
| --- | --- |
| 1 2 3 | doc1: [I, like, NLP]  doc2: [I, like, CS, and, like, NLP] |

**词集模型**

|  |  |
| --- | --- |
| 1 2 | doc1: [1, 0, 1, 1, 0] doc2: [1, 1, 1, 1, 1] |

**词袋模型**

|  |  |
| --- | --- |
| 1 2 | doc1: [1, 0, 1, 1, 0] doc2: [1, 1, 2, 1, 1] # like出现了两次故词频为2 |

##### tf-idf

