Atitit nlp文本挖掘和自然语言处理方面，常用的算法总结 比如tf-idf

目录

[1.1. tf：词频，是指某个词在某篇文章中出现的频率 2](#_Toc19863)

[1.2. 去停用词算法 2](#_Toc13519)

[1.3. idf。 3](#_Toc3812)

[1.4. 分词算法 5](#_Toc8092)

[1.5. 关键词提取 5](#_Toc26111)

[1.6. 摘要算法textbank算法 5](#_Toc26850)

[1.7. 参考《文本相似度-bm25算法原理及实现》 5](#_Toc24326)

[1.8.  Kmeans聚类 5](#_Toc15715)

[1.9. 基于改进编辑距离的字符串相似度求解算法 5 新闻中完整性对象的识别 6 基于互信息和左右信息熵的短语提取 5](#_Toc26725)

[1.10.  C-value、D-value算法 5](#_Toc15295)

[1.11. 文本标签路径比抽取新闻网页正文、时间、标题等信息 5](#_Toc4219)

[1.12. MaxMatch文本匹配算法其实是一种基于贪婪策略的算法 6](#_Toc539)

[2. 分词算法 6](#_Toc2353)

[2.1. 第一种是基于字典的分词，先把句子按照字典切分成词，再寻找词的最佳组合方式； 6](#_Toc7619)

[2.2. 第二种是基于字的分词，即由字构词，先把句子分成一个个字，再将字组合成词，寻找最优的切分策略， 6](#_Toc27004)

[3. 2. 分词算法 3 7](#_Toc5422)

[3.1. 2.1 基于词典的分词 4 7](#_Toc9202)

[3.2. 2.1.1 最大匹配分词算法 4 7](#_Toc18123)

[3.3. 2.1.2 最短路径分词算法 4 7](#_Toc8901)

[3.4. 2.1.3 基于n-gram model的分词算法 6 7](#_Toc27512)

[3.5. 2.2 基于字的分词 6 7](#_Toc15594)

[3.6. 2.2.1 生成式模型分词算法 7 7](#_Toc16803)

[3.7. 2.2.2 判别式模型分词算法 8 7](#_Toc5435)

[3.8. 2.2.3 神经网络分词算法 8 7](#_Toc20055)

[3.9. 7](#_Toc15308)

[4. ref 7](#_Toc19394)

在文本挖掘和自然语言处理方面，tf-idf是非常重要也非常常用的算法

****TFIDF的主要思想是：如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类****

## tf：词频，是指某个词在某篇文章中出现的频率

。比如，某篇文章共1000个词汇，其中hello出现5次，那么其tf=5/1000。tf最直观的理解就是，当一个词在本文中出现的频率越高，则这篇文章的主题和这个词的相关可能性越大。

## 去停用词算法

这种直观理解是否准确呢？可以说相当不准确。举例来说，一篇文章中出现最多的字词可能是你、我、他、的、是、这、那等等。通过这些词来分析一篇文章的内涵几乎是不可能的。所以人们又做了进一步处理。就是把这些在每篇文章里都可能大量出现又和文章意义关联不大的词都去掉。这类词也有了一个专有名称：停用词。所以文本处理的前几步通常都包括这一步：去停用词，既能减少词汇处理量，又能有效减少歧义。属于重要的预处理步骤。

去掉停用词之后的词频是否就能比较准确的表达文章含义了呢？还是不够的。设想一下，如果一篇文章是描述一份国内专利的。文章里反复提到了“中国”两个字。中国这、中国那，结果中国这个词的词频最高，那么这个词和实际要说的专利有多大关联呢？基本没有。但是我们又不能把中国加到停用词里，否则停用词列表就太多了，而且去掉也不合理，万一某篇文章就是介绍中国的呢。这个时候就又发明另外一个算法：idf。

## idf。

TF-IDF（term frequency–inverse document frequency）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

TF-IDF加权的各种形式常被搜寻引擎应用

作为文件与用户查询之间相关程度的度量或评级。

除了TF-IDF以外，因特网上的搜寻引擎还会使用基于链接分析的评级方法，以确定文件在搜寻结果中出现的顺序：PR。

---------------------

 idf叫做反文档频率：目的就是针对刚才说的这种情况进行识别。还以上面为例，这篇文章中“中国”这个词的词频最高，却不能反应真实的文章内涵，这是为什么呢？很大程度是因为“中国”这个词太常见了，不仅在这篇文章里出现次数多，在其他文章里出现的次数也很多。这么一来，说明这个词不足以描述文章“特征”。于是评价某个词的“独特性”的公式idf就这样设计出来：语料库文章总数/包含某个词的文章数。意味着，如果一个词在越多的文章中出现过，那么其“独特性”就越低。出现的文章数越少，idf值越大，其独特性越高。整体思路就是这样，后续再加上一点数学上的处理：如果一个词在所有语料库的文章中都没出现过。那么分母就是0了，这在计算中会发生错误，所以往往把分母+1，保证其至少不会是0。虽然缺失了一点点精确性，但保证计算过程不至于出错。而且在语料库文章数量很大时，对结果的影响是微乎其微的。另外，这个除法除出来的结果可能差别很大，有的接近1（几乎每篇文章都出现这个词）、有的非常大（极少出现），这时候看起来值的差距太悬殊，不易计算也不易比较。于是再取一下对数。所以整个公式就是：

---------------------

最后，再把tf和idf相乘，这个得出来的值就很能反映文章的主题了。举个例子，现有文章库100万篇。当对一篇新来的文章进行分析时，发现其tf排名第一的是“中国”，30/1000(假定去掉停用词之后还有1000个词汇)=0.03，tf排名第二的是“青蒿素”，20/1000=0.02，再继续计算idf，发现包含“中国”的文章有10万篇，其idf=log(1000000/100000)=1；含有“青蒿素”的文章只有1000篇，其idf=log(1000000/1000)=3，最后“中国”的tf-idf值为0.03，“青蒿素”的tf-idf值为0.06。这样，如果按tf-idf值排序，尽管“中国”出现的次数多，但仍被排到“青蒿素”之后。说明这篇文章和青蒿素相关的可能性较大。当我们选取tf-idf值排名前若干的词汇作为一篇文章的主旨，可靠性就相对准确多了

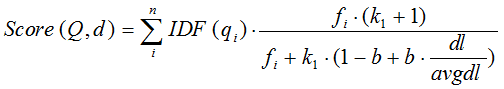
---------------------

## 分词算法

## 关键词提取

## 摘要算法textbank算法

## [参考《文本相似度-bm25算法原理及实现》](http://www.jianshu.com/p/1e498888f505" \t "https://blog.csdn.net/sinat_26917383/article/details/_blank)

BM25是通过q和s中的公共词汇进行相似度计算的算法,BM25算法的相关性得分公式可总结为：  


## Kmeans聚类

## 相似度算法bm25

## 文本分类 （贝叶斯算法

## 推荐算法

## 基于改进编辑距离的字符串相似度求解算法 5 新闻中完整性对象的识别 6 基于互信息和左右信息熵的短语提取

## **[C-value、D-value算法](https://www.jianshu.com/p/5395b09915bf" \t "https://www.jianshu.com/p/_blank)**

## 文本标签路径比抽取新闻网页正文、时间、标题等信息

## MaxMatch文本匹配算法其实是一种基于贪婪策略的算法

# 分词算法

我认为分词算法根据其核心思想主要分为两种，

## 第一种是基于字典的分词，先把句子按照字典切分成词，再寻找词的最佳组合方式；

## 第二种是基于字的分词，即由字构词，先把句子分成一个个字，再将字组合成词，寻找最优的切分策略，

同时也可以转化成序列标注问题。归根结底，上述两种方法都可以归结为在图或者概率图上寻找最短路径的问题。接下来将以“他说的确实在理”这句话为例，讲解各个不同的分词算法核心思想。

---------------------

# 2. 分词算法 3

## 2.1 基于词典的分词 4

## 2.1.1 最大匹配分词算法 4

## 2.1.2 最短路径分词算法 4

## 2.1.3 基于n-gram model的分词算法 6

## 2.2 基于字的分词 6

## 2.2.1 生成式模型分词算法 7

## 2.2.2 判别式模型分词算法 8

## 2.2.3 神经网络分词算法 8

## 

# ref

(9+条消息)tf-idf（term frequency–inverse document frequency）含义 - zjc的专栏 - CSDN博客.html