

# python+gensim | jieba分词、词袋doc2bow、TFIDF文本挖掘

原创 2017年05月08日 22:24:21

- 标签：
- doc2bow (<http://so.csdn.net/so/search/s.do?q=doc2bow&t=blog>) /
- tfidf (<http://so.csdn.net/so/search/s.do?q=tfidf&t=blog>) /
- jieba (<http://so.csdn.net/so/search/s.do?q=jieba&t=blog>) /
- gensim (<http://so.csdn.net/so/search/s.do?q=gensim&t=blog>) /
- python (<http://so.csdn.net/so/search/s.do?q=python&t=blog>) /
- 4191
- 编辑 (<http://write.blog.csdn.net/postedit/71436563>)
- 删除

分词这块之前一直用R在做，R中由两个jiebaR+Rwordseg来进行分词，来看看python里面的jieba.

之前相关的文章：

R语言 | 文本挖掘之中文分词包——Rwordseg包(原理、功能、详解)

([/sinat\\_26917383/article/details/51056068](/sinat_26917383/article/details/51056068))

R语言 | 文本挖掘——jiabaR包与分词向量化的simhash算法（与word2vec简单比较）

([/sinat\\_26917383/article/details/51068097](/sinat_26917383/article/details/51068097))

.

---

## 一、jieba分词功能

来源github：<https://github.com/fxsjy/jieba> (<https://github.com/fxsjy/jieba>)

### 1、主要模式

支持三种分词模式：

- 精确模式，试图将句子最精确地切开，适合文本分析；
- 全模式，把句子中所有的可以成词的词语都扫描出来，速度非常快，但是不能解决歧义；
- 搜索引擎模式，在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。

支持繁体分词

支持自定义词典

.

## 2、算法

- 基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图 (DAG)
- 采用了动态规划查找最大概率路径，找出基于词频的最大切分组合
- 对于未登录词，采用了基于汉字成词能力的 HMM 模型，使用了 Viterbi 算法

.

## 3、主要功能

- jieba.cut

方法接受三个输入参数: 需要分词的字符串; cut\_all 参数用来控制是否采用全模式; HMM 参数用来控制是否使用 HMM 模型

- jieba.cut\_for\_search

方法接受两个参数：需要分词的字符串；是否使用 HMM

模型。该方法适合用于搜索引擎构建倒排索引的分词，粒度比较细 待分词的字符串可以是 unicode 或 UTF-8 字符串、字符串。注意：不建议直接输入 GBK 字符串，可能无法预料地错误解码成 UTF-8

- jieba.Tokenizer(dictionary=DEFAULT\_DICT)

新建自定义分词器，可用于同时使用不同词典。jieba.dt 为默认分词器，所有全局分词相关函数都是该分词器的映射。

- 载入词典

用法：jieba.load\_userdict(file\_name) # file\_name 为文件类对象或自定义词典的路径

词典格式和 dict.txt

一样，一个词占一行；每一行分三部分：词语、词频（可省略）、词性（可省略），用空格隔开，顺序不可颠倒。file 若为路径或二进制方式打开的文件，则文件必须为 UTF-8 编码。词频省略时使用自动计算的能保证分出该词的词频。

- 调整词典。

使用 add\_word(word, freq=None, tag=None) 和 del\_word(word)

可在程序中动态修改词典。

使用 suggest\_freq(segment, tune=True) 可调节单个词语的词频，使其能（或不能）被分出来。

注意：自动计算的词频在使用 HMM 新词发现功能时可能无效。

- 基于 TF-IDF 算法的关键词抽取

```
import jieba.analyse
```

```
jieba.analyse.extract_tags(sentence, topK=20, withWeight=False, allowPOS=())
```

sentence 为待提取的文本

topK 为返回几个 TF/IDF 权重最大的关键词，默认值为 20

withWeight 为是否一并返回关键词权重值，默认值为 False

allowPOS 仅包括指定词性的词，默认值为空，即不筛选

jieba.analyse.TFIDF(idf\_path=None) 新建 TFIDF 实例，idf\_path 为 IDF 频率文件

- 基于 TextRank 算法的关键词抽取

```
jieba.analyse.textrank(sentence, topK=20, withWeight=False, allowPOS=('ns', 'n', 'vn', 'v')) 直接使用，接口相同，注意!
```

```
jieba.analyse.TextRank() 新建自定义 TextRank 实例
```

- 词性标注

```
jieba.posseg.POSTokenizer(tokenizer=None) 新建自定义分词器，tokenizer 参数可指定内部使用的 jieba.Tokenizer 分词器  
标注句子分词后每个词的词性，采用和 ictclas 兼容的标记法。
```

- 并行分词

基于 python 自带的 multiprocessing 模块，目前暂不支持 Windows

用法：

```
jieba.enable_parallel(4) # 开启并行分词模式，参数为并行进程数
```

```
jieba.disable_parallel() # 关闭并行分词模式
```

.

## 二、gensim的doc2bow实现词袋模型

词袋模型不做过多介绍，直接来个案例

```
from gensim import corpora, models, similarities

raw_documents = [
    '0无偿居间介绍买卖毒品的行为应如何定性',
    '1吸毒男动态持有大量毒品的行为该如何认定',
    '2如何区分是非法种植毒品原植物罪还是非法制造毒品罪',
    '3为毒贩卖毒品提供帮助构成贩卖毒品罪',
    '4将自己吸食毒品原价转让给朋友吸食的行为该如何认定',
    '5为获报酬帮人购买毒品的行为该如何认定',
    '6毒贩出狱后再次够买毒品途中被抓的行为认定',
    '7虚夸毒品功效劝人吸食毒品的行为该如何认定',
    '8妻子下落不明丈夫又与他人登记结婚是否为无效婚姻',
    '9一方未签字办理的结婚登记是否有效',
    '10夫妻双方1990年按农村习俗举办婚礼没有结婚证 一方可否起诉离婚',
    '11结婚前对方父母出资购买的住房写我们二人的名字有效吗',
    '12身份证被别人冒用无法登记结婚怎么办?',
    '13同居后又与他人登记结婚是否构成重婚罪',
    '14未办登记只举办结婚仪式可起诉离婚吗',
    '15同居多年未办理结婚登记，是否可以向法院起诉要求离婚'
]
```

载入中文数据以及对应的包，corpora是构造词典的，similarities求相似性可以用得到。

```
texts = [[word for word in jieba.cut(document, cut_all=True)] for document in raw_documents]
```

将词语进行分词，并进行存储。

```
dictionary = corpora.Dictionary(texts)
```

寻找整篇语料的词典、所有词，corpora.Dictionary。

```
corpus = [dictionary.doc2bow(text) for text in texts]
```

建立语料之后，分支一：BOW词袋模型；分支二：建立TFIDF。

.

## 分之一：BOW词袋模型

由doc2bow变为词袋，输出的格式为：

```
[[[0, 1), (1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6, 1)], [(0, 1), (4, 1), (5, 1), (7, 1), (8, 1), (9, 2), (10, 1)], [(0, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 2), (10, 1)], [(0, 1), (3, 1), (4, 1), (5, 1), (6, 1), (7, 1), (8, 1), (9, 2), (10, 1)]]
```

例如（9，2）这个元素代表第二篇文档中id为9的单词“silver”出现了2次。

.

## 分支二：建立TFIDF

```
tfidf = models.TfidfModel(corpus)
```

使用tf-idf 模型得出该评论集的tf-idf 模型

```
corpus_tfidf = tfidf[corpus]
```

此处已经计算得出所有评论的tf-idf 值

在TFIDF的基础上，进行相似性检索。

```
similarity = similarities.Similarity('Similarity-tfidf-index', corpus_tfidf, num_features=600)
```

然后进行similarity检索。

```
print(similarity[test_corpus_tfidf_1]) # 返回最相似的样本材料,(index_of_document, similarity) tuples
```

当然其中的test\_corpus\_tfidf\_1需要进行预先处理。先变为dow2bow，然后tfidf

## 情况一：新的句子

```
new_sence = "16通过下面一句得到语料中每一篇文章对应的稀疏向量"  
test_corpus_1 = dictionary.doc2bow(jieba.cut(raw_documents[1], cut_all=True))  
vec_tfidf = tfidf[test_corpus_1]
```

利用doc2bow对其进行分割，然后求tfidf模型。输出的结果即为：

```
vec_tfidf  
Out[82]:  
[(1, 0.09586155438319434),  
 (5, 0.1356476941913782),  
 (6, 0.09586155438319434),  
 (8, 0.1356476941913782),  
 (11, 0.19172310876638868),  
 (12, 0.38344621753277736),  
 (13, 0.38344621753277736),  
 (14, 0.38344621753277736),  
 (15, 0.16086258119086566),  
 (16, 0.38344621753277736),  
 (17, 0.38344621753277736),  
 (18, 0.38344621753277736)]
```

## 情况二：tfidf模型的保存与内容查看

```
for item in corpus_tfidf:
    print(item)
tfidf.save("data.tfidf")
tfidf = models.TfidfModel.load("data.tfidf")
print(tfidf_model.dfs)
```

[阅读全文](#)

版权声明：本文为博主原创文章，转载请注明来源“素质云博客”，谢谢合作！！微信公众号：素质云笔记

### 相关文章推荐

## 基于gensim的Doc2Vec简析 (/lenbow/article/details/52120230)

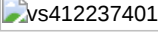
摘要：本文主要描述了一种文章向量（doc2vec）表示及其训练的相关内容，并列出了相关例子。两位大牛Quoc Le 和 Tomas Mikolov（搞出Word2vec的家伙）在2014年的《Distr...

-  (/lenbow)
- lenbow (/lenbow)
- 2016年08月04日 16:10
- 24083

## 用docsim/doc2vec/LSH比较两个文档之间的相似度 (/vs412237401/article/details/52238248)



在我们做文本处理的时候，经常需要对两篇文档是否相似做处理或者根据输入的文档，找出最相似的文档。幸好gensim提供了这样的工具，具体的处理思路如下，对于中文文本的比较，先需要做分词处理，根据分词的结...

-  vs412237401 (/vs412237401)
- vs412237401 (/vs412237401)
- 2016年08月18日 10:27
- 10215



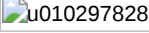
## 2017年前端报告：程序员薪酬上涨70%！

前端程序员的薪酬曝光，2017年，平均上涨70%，月薪20的人最为常见！以下为详细数据.....

([http://www.baidu.com/cb.php?c=lgF\\_pyfqHmknj0dP1f0IZ0qnfK9ujYzP1nYPH0k0Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1Y4nyDYn1RYmvcYPWN9uyNB0AwY5HDdnHnYnjc3nWR0IgF\\_5y9YIZ0IQzq-uZR8mLPbUB48ugfElAqspynElvNBnHqdlAdxTvqdThP-5yF\\_UvTkn0KzujYk0AFV5H00TZcq0KdpyfqHRLPjnvnfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnHbYPWn](http://www.baidu.com/cb.php?c=lgF_pyfqHmknj0dP1f0IZ0qnfK9ujYzP1nYPH0k0Aw-5Hc3rHnYnHb0TAq15HfLPWRznjb0T1Y4nyDYn1RYmvcYPWN9uyNB0AwY5HDdnHnYnjc3nWR0IgF_5y9YIZ0IQzq-uZR8mLPbUB48ugfElAqspynElvNBnHqdlAdxTvqdThP-5yF_UvTkn0KzujYk0AFV5H00TZcq0KdpyfqHRLPjnvnfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnHbYPWn))


## 基于gensim的文本主题模型(LDA)分析 (/u010297828/article/details/50464845)

主题模型文本分析小例子

-  (/u010297828)
- u010297828 (/u010297828)
- 2016年01月05日 20:56
- 7070

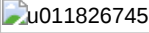
## gensim试用 (/largetalk/article/details/10439459)

gensim试用 gensim: <http://radimrehurek.com/gensim/index.html> Gensim is a free Python framework d...

-  (/largetalk)
- largetalk (/largetalk)
- 2013年08月28日 12:22
- 22711

## gensim similarity计算文档相似度 (/u011826745/article/details/52459891)

向量空间模型计算文档集合相似性。将原始输入的词转换为ID，词的id表示法简单易用，但是无法预测未登记词，难以挖掘词关系；词汇鸿沟[1]:任意两个词之间是独立的，无法通过词的ID来判断词语之间的关系...

-  (/u011826745)
- u011826745 (/u011826745)
- 2016年09月07日 15:15
- 3882



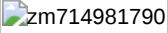
## 一学就会的 WordPress 实战课

认真学完这个系列课程之后，会深入了解 WordPress 的使用和开发，并掌握基本的 WordPress 的开发能力，后续可以根据需要开发适合自己的主题、插件，打造最个性的 WordPress 站点。

([http://www.baidu.com/cb.php?c=lgF\\_pyfqhHmknjTYPHf0IZ0qnfK9ujYzP1f4Pjnd0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1Y3myNhuANWnhR4uWP9mWDk0AwY5HDdnHnYnjc3nWm0lgF\\_5y9YIZ0lQzqMpgwBUvqoQhP8QviGIAPCmgfEmvq\\_lyd8Q1N9nHmV5HDznHN9mhkEusKzujYk0AFV5H00TZcqn0KdpyfqhHRLPjnvnfKEpyfqhHnsnj0YnsKWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnWn3PH0](http://www.baidu.com/cb.php?c=lgF_pyfqhHmknjTYPHf0IZ0qnfK9ujYzP1f4Pjnd0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1Y3myNhuANWnhR4uWP9mWDk0AwY5HDdnHnYnjc3nWm0lgF_5y9YIZ0lQzqMpgwBUvqoQhP8QviGIAPCmgfEmvq_lyd8Q1N9nHmV5HDznHN9mhkEusKzujYk0AFV5H00TZcqn0KdpyfqhHRLPjnvnfKEpyfqhHnsnj0YnsKWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnWn3PH0))

## 自然语言处理——简单词袋模型 (/zm714981790/article/details/51304506)

What Is Natural Language Processing? 本文将学习自然语言处理，当给予计算机一篇文章，它并不知道这篇文章的含义。为了让计算机可以从文章中做出推断，我们需要将文章转...

-  (/zm714981790)
- zm714981790 (/zm714981790)
- 2016年05月03日 16:12
- 2532

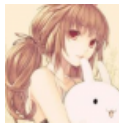
## BoW(词袋模型)+python代码实现 (/lilai619/article/details/46740837)

文章是参考整理得来，大家可以去文章最后的参考文献中去查看原文！文档主要分以下几部分内容: 1.SIFT 2.BOW 3.VLAD（未包含，请直接在下面的资源链接...

-  (/lilai619)
- lilai619 (/lilai619)
- 2015年07月03日 13:50
- 4506

## Python与自然语言处理（二）基于Gensim的 Word2Vec (/monkey131499/article/details/51113629)

Java调用NLPIC实现分词和标注工作，Python调用Word2Vec实现词向量相似度计算



- (/monkey131499)
- monkey131499 (/monkey131499)
- 2016年04月12日 10:13
- 8524

## gensim使用方法以及例子 (/u014595019/article/details/52218249)


gensim是一个python的自然语言处理库，能够将文档根据TF-IDF, LDA, LSI 等模型转化成向量模式，以便进行进一步的处理。此外，gensim还实现了word2vec功能，能够将单词转...



- (/u014595019)
- u014595019 (/u014595019)
- 2016年08月16日 10:58
- 12397

## jieba结巴分词--关键词抽取（核心词抽取） (/suibianshen2012/article/details/68927060)

转自：http://www.cnblogs.com/zhbzz2007 欢迎转载，也请保留这段声明。谢谢！ 1 简介 关键词抽取就是从文本里面把跟这篇文档意义最相关的一些词抽取出来。这个可...

-  suibianshen2012 (/suibianshen2012)
- suibianshen2012 (/suibianshen2012)
- 2017年03月31日 16:57
- 3635



(/sinat\_26917383)

sinat\_26917383 (/sinat\_26917383)

+ 关注

原创

202

粉丝

751

喜欢

0

码云

## 他的最新文章

更多文章 (/sinat\_26917383)

- python | apple开源机器学习框架turicreate中的SFrame——新形态pd.DataFrame (/sinat\_26917383/article/details/78805714)
- docker | 在nvidia-docker中使用tensorflow-gpu/jupyter (/sinat\_26917383/article/details/78728215)
- python | imagehash中的四种图像哈希方式 ( phash/ahash/dhash/小波hash ) (/sinat\_26917383/article/details/78582064)
-

python | 利用dlib和opencv实现简单换脸、人脸对齐、关键点定位与画图  
(/sinat\_26917383/article/details/78564416)

python | matplotlib使用（读入、显示、写出、opencv混用、格式转换...）  
(/sinat\_26917383/article/details/78559709)

相关推荐

- 基于gensim的Doc2Vec简析 (/lenbow/article/details/52120230)
- 用docsim/doc2vec/LSH比较两个文档之间的相似度 (/vs412237401/article/details/52238248)
- 基于gensim的文本主题模型(LDA)分析 (/u010297828/article/details/50464845)
- gensim试用 (/largetalk/article/details/10439459)



领取牛股



app外包公司

tpo听力文本

图形工作站

杨百万股票博客

standalone

美国留学博客

英语人机对话

博主专栏



(/column/details/13587.html)

R的数据操作与清洗 (/column/details/13587.html)

250448

•



21

(/column/details/13670.html)

R语言与自然语言处理 (/column/details/13670.html)

149280

展开

## 在线课程



• (http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjmsnjD0IZ0qnfK9ujYzP1mznWR10Aw-

5Hc4n1RLPWT0TAq15HR1rjfk100T1Y3PHf1mWdnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YIZ0IQzq-

uZR8mLPbUB48ugfEIAqspynETZ-

YpAq8n6KzujYk0AFV5H00TZcqn0KdpyfqHRLPjnvfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnHf1rHR)

Python全栈工程师 (http://www.baidu.com/cb.php?c=lgF\_pyfqHmknjmsnjD0IZ0qnfK9ujYzP1mznWR10Aw-

5Hc4n1RLPWT0TAq15HR1rjfk100T1Y3PHf1mWdnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YIZ0IQzq-

uZR8mLPbUB48ugfEIAqspynETZ-

YpAq8n6KzujYk0AFV5H00TZcqn0KdpyfqHRLPjnvfKEpyfqHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqnHf1rHR)

作者：韦玮



• (http://www.baidu.com/cb.php?c=lgF\_pyfqnHmknjmsnjc0lZ0qnfK9ujYzP1mznWR10Aw-

5Hc4n1RLPWT0TAq15HR1rjfkN100T1Y3PHf1mWndnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YlZ0lQzq-uZR8mLPbUB48ugfElAqspynEmybz0ZFb5HD0mhYqn0KsTWYs0ZNGujYkPHTYn1mk0AqGujYknWb3rjDY0APGujYLnWm4n1c0ULI85H00TZbqnW0v0APzm1Y1P人工智能工程师直通车 (http://www.baidu.com/cb.php?

c=lgF\_pyfqnHmknjmsnjc0lZ0qnfK9ujYzP1mznWR10Aw-5Hc4n1RLPWT0TAq15HR1rjfkN100T1Y3PHf1mWndnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YlZ0lQzq-uZR8mLPbUB48ugfElAqspynEmybz0ZFb5HD0mhYqn0KsTWYs0ZNGujYkPHTYn1mk0AqGujYknWb3rjDY0APGujYLnWm4n1c0ULI85H00TZbqnW0v0APzm1Y1P

作者：卿来云



• (http://www.baidu.com/cb.php?c=lgF\_pyfqnHmknjmsnjn0lZ0qnfK9ujYzP1mznWR10Aw-

5Hc4n1RLPWT0TAq15HR1rjfkN100T1Y3PHf1mWndnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YlZ0lQzq-uZR8mLPbUB48ugfElAqspynElvNBn6KzujYk0AFV5H00TZcqn0KdpyfqnHRLPjnvnfKEpyfqnHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqPHf3P6)Web前端工程师 (http://www.baidu.com/cb.php?c=lgF\_pyfqnHmknjmsnjn0lZ0qnfK9ujYzP1mznWR10Aw-

5Hc4n1RLPWT0TAq15HR1rjfkN100T1Y3PHf1mWndnWmknhRkPH6v0AwY5HDdnHnYnjc3nWm0lgF\_5y9YlZ0lQzq-uZR8mLPbUB48ugfElAqspynElvNBn6KzujYk0AFV5H00TZcqn0KdpyfqnHRLPjnvnfKEpyfqnHc4rj6kP0KWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqPHf3P6)

作者：曾亮

他的热门文章

- R语言 | list用法、批量读取、写出数据时的用法 (/sinat\_26917383/article/details/51123214) 34517
- R语言与格式、日期格式、格式转化 (/sinat\_26917383/article/details/50677065) 29063
- R语言 | 决策树族——随机森林算法 (/sinat\_26917383/article/details/51308061) 27872
- R语言数据集合并、数据增减、不等长合并 (/sinat\_26917383/article/details/50676894) 21646
- R语言 | 文本挖掘——词云wordcloud2包 (/sinat\_26917383/article/details/51620019) 20812



- 
- 
- 

内容举报  
返回顶部  
收藏助手

不良信息举报

您举报文章：python+gensim | jieba分词、词袋doc2bow、TFIDF文本挖掘 (/sinat\_26917383/article/details/71436563)

举报原因：☐色情 ☐政治 ☐抄袭 ☐广告 ☐招聘 ☐骂人

因：☐其他

原因补充：

(最多只允许输入30个字)

