

(/apps/redirect?
utm_source=side-
banner-click)

python文本相似度计算



lyy0905 (/u/7cfbe2cc1491) [+关注](#)

2017.04.28 22:27* 字数 1433 阅读 33416 评论 12 喜欢 55 阅读 33416 评论 12 喜欢 55 (/u/7cfbe2cc1491)

步骤

1. 分词、去停用词
2. 词袋模型向量化文本
3. TF-IDF模型向量化文本
4. LSI模型向量化文本
5. 计算相似度

理论知识

两篇中文文本，如何计算相似度？相似度是数学上的概念，自然语言肯定无法完成，所有要把文本转化为向量。两个向量计算相似度就很简单的，欧式距离、余弦相似度等等各种方法，只需要中学水平的数学知识。

那么如何将文本表示成向量呢？

• 词袋模型

最简单的表示方法是词袋模型。把一篇文本想象成一个个词构成的，所有词放入一个袋子里，没有先后顺序、没有语义。

例如：

John likes to watch movies. Mary likes too.

John also likes to watch football games.

这两个句子，可以构建出一个词典，key为上文出现过的词，value为这个词的索引序号

```
{"John": 1, "likes": 2, "to": 3, "watch": 4, "movies": 5, "also": 6, "football": 7, "games": 8, "Mary": 9, "too": 10}
```

那么，上面两个句子用词袋模型表示成向量就是：

```
[1, 2, 1, 1, 1, 0, 0, 0, 1, 1]
```

```
[1, 1, 1, 1, 0, 1, 1, 1, 0, 0]
```

相对于英文，中文更复杂一些，涉及到分词。准确地分词是所有中文文本分析的基础，本文使用结巴分词，完全开源而且分词准确率相对有保障。

• TF-IDF模型

词袋模型简单易懂，但是存在问题。中文文本里最常见的词是“的”、“是”、“有”这样的没有实际含义的词。一篇关于足球的中文文本，“的”出现的数量肯定多于“足球”。所以，要对文本中出现的词赋予权重。

一个词的权重由 $TF * IDF$ 表示，其中TF表示词频，即一个词在这篇文本中出现的频率；IDF表示逆文档频率，即一个词在所有文本中出现的频率倒数。因此，一个词在某文本中出现的越多，在其他文本中出现的越少，则这个词能很好地反映这篇文本的内容，权重就越大。

广告

(https://dsp-
click.youdao.com/clk/re
slot=30edd91dd8637d
65d4-4bf3-9fc8-
87fa848f0feb&iid=%7E
832551699796901624



回过头看词袋模型，只考虑了文本的词频，而TF-IDF模型则包含了词的权重，更加准确。文本向量与词袋模型中的维数相同，只是每个词的对应分量值换成了该词的TF-IDF值。

$$\text{TF}(t, d) = \frac{\text{词 } t \text{ 在文档 } d \text{ 中出现的次数}}{\text{文档 } d \text{ 的总词数}}$$

TF

$$\text{IDF}(t) = \log \frac{\text{语料库中文档总数}}{\text{包含词 } t \text{ 的文档数} + 1}$$

IDF

- LSI模型

TF-IDF模型足够胜任普通的文本分析任务，用TF-IDF模型计算文本相似度已经比较靠谱了，但是细究的话还存在不足之处。实际的中文文本，用TF-IDF表示的向量维数可能是几百、几千，不易分析计算。此外，一些文本的主题或者说中心思想，并不能很好地通过文本中的词来表示，能真正概括这篇文本内容的词可能没有直接出现在文本中。

因此，这里引入了Latent Semantic Indexing (LSI) 从文本潜在的主题来进行分析。LSI是概率主题模型的一种，另一种常见的是LDA，核心思想是：每篇文本中有多个概率分布不同的主题；每个主题中都包含所有已知词，但是这些词在不同主题中的概率分布不同。LSI通过奇异值分解的方法计算出文本中各个主题的概率分布，严格的数学证明需要看相关论文。假设有5个主题，那么通过LSI模型，文本向量就可以降到5维，每个分量表示对应主题的权重。

python实现

分词上使用了结巴分词 (<https://link.jianshu.com?t=https://github.com/fxsjy/jieba>)，词袋模型、TF-IDF模型、LSI模型的实现使用了gensim (<https://link.jianshu.com?t=https://github.com/RaRe-Technologies/gensim>)库。

```
import jieba.posseg as pseg
import codecs
from gensim import corpora, models, similarities
```

构建停用词表

```
stop_words = '/Users/yiiyuanliu/Desktop/nlp/demo/stop_words.txt'
stopwords = codecs.open(stop_words, 'r', encoding='utf8').readlines()
stopwords = [w.strip() for w in stopwords]
```

结巴分词后的停用词性[标点符号、连词、助词、副词、介词、时语素、‘的’、数词、方位词、代词]

```
stop_flag = ['x', 'c', 'u', 'd', 'p', 't', 'uj', 'm', 'f', 'r']
```

对一篇文章分词、去停用词

```
def tokenization(filename):  
    result = []  
    with open(filename, 'r') as f:  
        text = f.read()  
        words = pseg.cut(text)  
    for word, flag in words:  
        if flag not in stop_flag and word not in stopwords:  
            result.append(word)  
    return result
```

选取三篇文章，前两篇是高血压主题的，第三篇是iOS主题的。

```
filenames = ['/Users/yiiyuanliu/Desktop/nlp/demo/articles/13 件小事帮您稳血压  
.txt',  
             '/Users/yiiyuanliu/Desktop/nlp/demo/articles/高血压患者宜喝低脂奶  
.txt',  
             '/Users/yiiyuanliu/Desktop/nlp/demo/articles/ios.txt'  
            ]  
corpus = []  
for each in filenames:  
    corpus.append(tokenization(each))  
print len(corpus)
```

```
Building prefix dict from the default dictionary ...  
Loading model from cache /var/folders/1q/5404x10d3k76q2wqys68pzkh0000gn/T/j  
ieba.cache  
Loading model cost 0.349 seconds.  
Prefix dict has been built successfully.
```

3

建立词袋模型

```
dictionary = corpora.Dictionary(corpus)  
print dictionary
```

```
Dictionary(431 unique tokens: [u'\u627e\u51fa', u'\u804c\u4f4d', u'\u6253\u  
9f3e', u'\u4eba\u7fa4', u'\u996e\u54c1']...)
```

```
doc_vectors = [dictionary.doc2bow(text) for text in corpus]  
print len(doc_vectors)  
print doc_vectors
```

```

3
[[ (0, 1), (1, 3), (2, 2), (3, 1), (4, 3), (5, 3), (6, 3), (7, 1), (8, 1), (
9, 1), (10, 1), (11, 3), (12, 1), (13, 2), (14, 3), (15, 3), (16, 1), (17,
2), (18, 1), (19, 1), (20, 1), (21, 2), (22, 1), (23, 1), (24, 1), (25, 1),
(26, 1), (27, 3), (28, 1), (29, 1), (30, 1), (31, 1), (32, 1), (33, 1), (3
4, 1), (35, 1), (36, 1), (37, 1), (38, 1), (39, 1), (40, 2), (41, 1), (42,
2), (43, 1), (44, 2), (45, 1), (46, 4), (47, 1), (48, 2), (49, 1), (50, 2),
(51, 1), (52, 1), (53, 1), (54, 1), (55, 1), (56, 1), (57, 1), (58, 1), (5
9, 1), (60, 1), (61, 1), (62, 1), (63, 1), (64, 1), (65, 3), (66, 1), (67,
1), (68, 1), (69, 2), (70, 2), (71, 5), (72, 1), (73, 2), (74, 3), (75, 1),
(76, 1), (77, 1), (78, 2), (79, 1), (80, 1), (81, 1), (82, 1), (83, 2), (8
4, 3), (85, 1), (86, 2), (87, 1), (88, 3), (89, 1), (90, 1), (91, 1), (92,
2), (93, 1), (94, 1), (95, 2), (96, 2), (97, 1), (98, 3), (99, 1), (100, 1
), (101, 1), (102, 2), (103, 1), (104, 1), (105, 1), (106, 1), (107, 1), (10
8, 2), (109, 1), (110, 1), (111, 1), (112, 1), (113, 1), (114, 1), (115, 1)
, (116, 1), (117, 1), (118, 1), (119, 2), (120, 1), (121, 1), (122, 1), (12
3, 1), (124, 1), (125, 1), (126, 1), (127, 1), (128, 5), (129, 5), (130, 1)
, (131, 1), (132, 2), (133, 1), (134, 1), (135, 1), (136, 1), (137, 1), (13
8, 6), (139, 1), (140, 1), (141, 1), (142, 4), (143, 1), (144, 2), (145, 1)
, (146, 1), (147, 1), (148, 2), (149, 1), (150, 1), (151, 5), (152, 1), (15
3, 1), (154, 1), (155, 1), (156, 1), (157, 1), (158, 1), (159, 1), (160, 1)
, (161, 2), (162, 15), (163, 3), (164, 1), (165, 1), (166, 2), (167, 1), (1
68, 6), (169, 1), (170, 1), (171, 1), (172, 3), (173, 1), (174, 1), (175, 2
), (176, 1), (177, 1), (178, 2), (179, 2), (180, 1), (181, 6), (182, 1), (1
83, 1), (184, 1), (185, 2), (186, 1), (187, 1), (188, 1), (189, 1), (190, 1
), (191, 1), (192, 1), (193, 1), (194, 1), (195, 1), (196, 1), (197, 1), (1
98, 1), (199, 1), (200, 1), (201, 5), (202, 1), (203, 2), (204, 2), (205, 1
), (206, 1), (207, 1), (208, 1), (209, 2), (210, 1), (211, 1), (212, 1), (2
13, 1), (214, 1), (215, 1), (216, 1), (217, 1), (218, 1), (219, 3), (220, 1
), (221, 1), (222, 4), (223, 1), (224, 1), (225, 1), (226, 1), (227, 1), (2
28, 1), (229, 1), (230, 1), (231, 2), (232, 12), (233, 1), (234, 1), (235,
1), (236, 2), (237, 1), (238, 1), (239, 1), (240, 1), (241, 1), (242, 1), (
243, 1), (244, 1), (245, 1), (246, 1), (247, 4), (248, 2), (249, 1), (250,
1), (251, 1), (252, 1), (253, 2), (254, 1), (255, 1), (256, 1), (257, 6), (
258, 1), (259, 2)], [(6, 1), (7, 1), (11, 1), (14, 1), (15, 2), (27, 1), (4
7, 2), (71, 1), (78, 1), (92, 2), (101, 1), (106, 1), (112, 4), (121, 1), (
138, 6), (143, 1), (151, 2), (155, 1), (158, 1), (162, 4), (170, 2), (203,
1), (213, 1), (227, 1), (232, 7), (254, 2), (260, 1), (261, 1), (262, 1), (
263, 1), (264, 1), (265, 1), (266, 1), (267, 2), (268, 1), (269, 1), (270,
1), (271, 1), (272, 1), (273, 1), (274, 1), (275, 1), (276, 2), (277, 3), (
278, 1), (279, 1), (280, 1), (281, 1), (282, 1), (283, 1), (284, 1), (285,
1), (286, 2), (287, 1), (288, 3), (289, 1), (290, 1), (291, 1), (292, 2), (
293, 2), (294, 1), (295, 1), (296, 1), (297, 3), (298, 1), (299, 1), (300,
1), (301, 1), (302, 1)], [(14, 5), (19, 1), (22, 1), (25, 1), (27, 3), (77,
3), (89, 1), (103, 2), (132, 1), (137, 2), (147, 1), (161, 1), (169, 5), (
201, 2), (208, 2), (257, 1), (266, 1), (272, 1), (303, 2), (304, 2), (305,
1), (306, 6), (307, 1), (308, 2), (309, 2), (310, 1), (311, 2), (312, 1), (
313, 1), (314, 10), (315, 1), (316, 1), (317, 3), (318, 1), (319, 1), (320,
1), (321, 3), (322, 2), (323, 3), (324, 2), (325, 14), (326, 1), (327, 1),
(328, 3), (329, 1), (330, 1), (331, 2), (332, 6), (333, 2), (334, 3), (335
, 1), (336, 1), (337, 1), (338, 1), (339, 1), (340, 4), (341, 1), (342, 1),
(343, 1), (344, 3), (345, 1), (346, 1), (347, 1), (348, 1), (349, 1), (350
, 1), (351, 2), (352, 4), (353, 2), (354, 1), (355, 1), (356, 1), (357, 3),
(358, 1), (359, 14), (360, 1), (361, 1), (362, 1), (363, 1), (364, 2), (36
5, 1), (366, 1), (367, 1), (368, 4), (369, 1), (370, 1), (371, 1), (372, 1)
, (373, 1), (374, 1), (375, 1), (376, 2), (377, 1), (378, 1), (379, 1), (38
0, 1), (381, 2), (382, 1), (383, 4), (384, 1), (385, 2), (386, 1), (387, 1)
, (388, 2), (389, 1), (390, 1), (391, 1), (392, 2), (393, 1), (394, 1), (39
5, 2), (396, 1), (397, 1), (398, 2), (399, 1), (400, 1), (401, 2), (402, 1)
, (403, 3), (404, 2), (405, 1), (406, 1), (407, 2), (408, 1), (409, 2), (41
0, 1), (411, 2), (412, 2), (413, 1), (414, 1), (415, 1), (416, 1), (417, 1)
, (418, 1), (419, 5), (420, 1), (421, 1), (422, 1), (423, 3), (424, 1), (42
5, 1), (426, 1), (427, 1), (428, 1), (429, 1), (430, 6)]]

```

建立TF-IDF模型

```

tfidf = models.TfidfModel(doc_vectors)
tfidf_vectors = tfidf[doc_vectors]

```

```

print len(tfidf_vectors)
print len(tfidf_vectors[0])

```

```
3
258
```

构建一个**query**文本，是高血压主题的，利用词袋模型的字典将其映射到向量空间

```
query = tokenization('/Users/yiiyuanliu/Desktop/nlp/demo/articles/关于降压药的五个问题.txt')
```

```
query_bow = dictionary.doc2bow(query)
```

```
print len(query_bow)
print query_bow
```

```
35
[(6, 1), (11, 1), (14, 1), (19, 1), (25, 1), (28, 1), (38, 2), (44, 3), (50, 4), (67, 1), (71, 1), (97, 1), (101, 3), (105, 2), (137, 1), (138, 4), (148, 6), (151, 2), (155, 1), (158, 3), (162, 4), (169, 1), (173, 2), (203, 1), (232, 12), (236, 1), (244, 9), (257, 1), (266, 1), (275, 2), (282, 1), (290, 2), (344, 1), (402, 1), (404, 3)]
```

```
index = similarities.MatrixSimilarity(tfidf_vectors)
```

用**TF-IDF**模型计算相似度，相对于前两篇高血压主题的文本，**iOS**主题文本与**query**的相似度很低。可见**TF-IDF**模型是有效的，然而在语料较少的情况下，与同是高血压主题的文本相似度也不高。

```
sims = index[query_bow]
print list(enumerate(sims))
```

```
[(0, 0.28532028), (1, 0.28572506), (2, 0.023022989)]
```

构建**LSI**模型，设置主题数为**2**（理论上这两个主题应该分别为高血压和**iOS**）

```
lsi = models.LsiModel(tfidf_vectors, id2word=dictionary, num_topics=2)
```

```
lsi.print_topics(2)
```

```
[(0,
  u'0.286*"u9ad8\u8840\u538b" + 0.241*"u8840\u538b" + 0.204*"u60a3\u8005" + 0.198*"u559d" + 0.198*"u4f4e" + 0.198*"u8865\u9499" + 0.155*"u538b\u529b" + 0.155*"u852c\u83dc" + 0.132*"u542b\u9499" + 0.132*"u8840\u9499"'),
 (1,
  u'0.451*"iOS" + 0.451*"u5f00\u53d1" + 0.322*"u610f\u4e49" + 0.193*"u57f9\u8bad" + 0.193*"u9762\u8bd5" + 0.193*"u884c\u4e1a" + 0.161*"u7b97\u6cd5" + 0.129*"u9ad8\u8003" + 0.129*"u5e02\u573a" + 0.129*"u57fa\u7840"')]
```

```
lsi_vector = lsi[tfidf_vectors]
for vec in lsi_vector:
    print vec
```

```
[(0, 0.74917098831536277), (1, -0.0070559356931168236)]  
[(0, 0.74809557226254608), (1, -0.054041302062161914)]  
[(0, 0.045784366765220297), (1, 0.99846660199817183)]
```

在LSI向量空间中，所有文本的向量都是二维的

```
query = tokenization('/Users/yiiyuanliu/Desktop/nlp/demo/articles/关于降压药的五个问题.txt')  
query_bow = dictionary.doc2bow(query)  
print query_bow
```

```
[(6, 1), (11, 1), (14, 1), (19, 1), (25, 1), (28, 1), (38, 2), (44, 3), (50, 4), (67, 1), (71, 1), (97, 1), (101, 3), (105, 2), (137, 1), (138, 4), (148, 6), (151, 2), (155, 1), (158, 3), (162, 4), (169, 1), (173, 2), (203, 1), (232, 12), (236, 1), (244, 9), (257, 1), (266, 1), (275, 2), (282, 1), (290, 2), (344, 1), (402, 1), (404, 3)]
```

```
query_lsi = lsi[query_bow]  
print query_lsi
```

```
[(0, 7.5170080232286249), (1, 0.10900815862153138)]
```

```
index = similarities.MatrixSimilarity(lsi_vector)  
sims = index[query_lsi]  
print list(enumerate(sims))
```

```
[(0, 0.99971396), (1, 0.99625134), (2, 0.060286518)]
```

可以看到LSI的效果很好，一个高血压主题文本与前两个训练文本的相似性很高，而与iOS主题的第三篇训练文本相似度很低

参考资料:

Coursera: Text Mining and Analytics (<https://link.jianshu.com?t=https://www.coursera.org/learn/text-mining>)

阮一峰: TF-IDF与余弦相似性的应用 (一): 自动提取关键词 (<https://link.jianshu.com?t=http://www.ruanyifeng.com/blog/2013/03/tf-idf.html>)

如何计算两个文档的相似度 (<https://link.jianshu.com?t=http://www.52nlp.cn/category/%E6%8E%A8%E8%8D%90%E7%B3%BB%E7%BB%9F>)

小礼物走一走，来简书关注我

赞赏支持

自然语言处理 (/nb/11725928)

举报文章 © 著作权归作者所有



lyy0905 (/u/7cfbe2cc1491)

写了 18899 字, 被 123 人关注, 获得了 137 个喜欢
(/u/7cfbe2cc1491) 写了 18899 字, 被 123 人关注, 获得了 137 个喜欢

[+ 关注](#)

浙江大学生物医学工程 硕士在读 自然语言处理 iOS开发

[喜欢](#) | 55[更多分享](#)

下载简书 App ▶
随时随地发现和创作内容



(/apps/redirect?utm_source=note-bottom-click)



[登录 \(/sign-in?utm_source=desktop&utm_medium=not-signed-in-comment-form\)](#)

12条评论[只看作者](#)[按时间倒序](#) [按时间正序](#)

七海七海、 (/u/a024509ffeda)

7楼 · 2018.12.25 17:14

(/u/a024509ffeda)

博主您好, 我现在有一个很大的文件, 每一行 为一篇文章, 用这个文件建立了语料库。我现在需要用新的文章和文件中的每一篇文章做相似度比较怎么做

[赞](#) [回复](#)

一起来听雨 (/u/e93b3ca274d5)

6楼 · 2018.06.26 09:33

(/u/e93b3ca274d5)

楼主您好!

从未接触过这方面的, 最近自己也在看。。我现在有一个excel表格, 表格有多个字段, 数百条记录。现在我想计算这数百条记录的相似度。相似度高的就归为一条记录。咨询下您有啥方法没? 谢谢啦

[赞](#) [回复](#)

翼_6ecb (/u/13f05aa60a26)

5楼 · 2018.05.15 22:09

(/u/13f05aa60a26)

博主您好, 我在用TF-IDF模型计算相似度时, 用了一些文档建立了语料库, 当我用建立语料库的这些文档中的某一篇查询相似度时, 发现查询的这一篇和语料库中的同一篇的相似度很接近1但却不是1, 请问这是为什么

[赞](#) [回复](#)

lyy0905 (/u/7cfbe2cc1491): 可以打印出这两篇文档的向量对比一下, 看是不是完全一样

2018.05.16 09:02 [回复](#)[添加新评论](#)



优蜜 (/u/727d372ea00b)

3楼 · 2018.03.08 17:55

(/u/727d372ea00b)

小白，前来请教，LSI主题模型中主题的个数是自己加上去的？不能自动判断吗，刚接触不太明白

赞 回复

lyy0905 (/u/7cfbe2cc1491): @优蜜 (/users/727d372ea00b) 算法本身需要给定主题的数量

2018.03.09 09:29 回复

优蜜 (/u/727d372ea00b): @lyy0905 (/users/7cfbe2cc1491) 谢谢，有没有那种自动判断主题类型的算法？

2018.03.10 14:52 回复

lyy0905 (/u/7cfbe2cc1491): @优蜜 (/users/727d372ea00b) 据我所知好像没有...

2018.03.10 20:19 回复

添加新评论



插着扇子的石头 (/u/155c204e9ff5)

2楼 · 2017.04.28 22:34

(/u/155c204e9ff5)

真巧，我也研究过搜索算法。

你可以看看BM25算法（TF-IDF模型的升级版），实际项目中效果不错哦😊

赞 回复

lyy0905 (/u/7cfbe2cc1491): @插着扇子的石头 (/users/155c204e9ff5) 多谢指点😊

2017.04.29 11:11 回复

插着扇子的石头 (/u/155c204e9ff5): @lyy0905 (/users/7cfbe2cc1491) 相互学习，我研究生是生物信息方向，和你的专业还有点相似😊

2017.04.29 11:15 回复

lyy0905 (/u/7cfbe2cc1491): @插着扇子的石头 (/users/155c204e9ff5) 哈哈好巧 要向你多学习 感觉以前学得太杂了

2017.04.29 11:23 回复

添加新评论

被以下专题收入，发现更多相似内容

-  @IT·互联网 (/c/V2CqjW?utm_source=desktop&utm_medium=notes-included-collection)
-  数据乐园 (/c/a3017f6e996e?utm_source=desktop&utm_medium=notes-included-collection)
-  Machine... (/c/4b3f92364497?utm_source=desktop&utm_medium=notes-included-collection)
-  机器学习与数据挖掘 (/c/9ca077f0fae8?utm_source=desktop&utm_medium=notes-included-collection)
-  Python语... (/c/9bc3ae683403?utm_source=desktop&utm_medium=notes-included-collection)
-  机器学习 (/c/67f720cb74aa?utm_source=desktop&utm_medium=notes-included-collection)
-  机器学习 (/c/95735dac0399?utm_source=desktop&utm_medium=notes-included-collection)

展开更多 ∨

推荐阅读

更多精彩内容 > (/)

离开上海，到郑州漂泊的日子 (/p/e12ad67b33fc?utm_campaign=males...

此时此刻，我正在郑州的某个小区，借住在朋友家，用朋友的电脑，敲下这一个又一个忧伤的文字。自从离开上海，我就一直在奔波。从上海回家，从家里去姑姑家，从姑姑家到县城，从县城来到郑州。家里的

微光隐隐 (/u/1939c693d394?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

4个月零基础入门，一站式掌握AI红利！

硅谷大咖带你从零入门AI！报名立享900元新年专属福利！

广告

(https://dsp-click.youdao.com/clk/request.s?

slot=f2ac00aef0eb6b673f4e4639046bc6f8&k=ABh9tY866L4%2FqxpRqpwhR3ht8%2BEtetxY%2FMa7EAEwDj)2726-4469-b764-e4226bfdad62&iid=%7B%22-1917143398108168935%22%3A1%7D&sid=17836)

微小说|凑巧 (/p/2b57f194770d?utm_campaign=ma... (/p/2b57f194770d?

2019年1月15日，星期二，天气晴朗 今天，我们几个高中同学聚会，喝到面热酒酣之际，又说起范范和果果的事情来。我们要求范范老实交代，当年是怎么

11山山 (/u/f627d3b0534e?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

这一刻，我不想喊您无戒老师 (/p/8e2fb9f93e43?ut... (/p/8e2fb9f93e43?

原创文字，简书首发，谢绝转载！ 今年三月份，我报了无戒老师的90天训练营，此后，我一路追随，再也没离开。老实说，我是无戒老师训练营里最不起

暖暖的冬花 (/u/64acbf6b656d?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

喜欢写故事小说，想投稿赚钱，你就进来吧！ (/p/4100102f0149?utm_ca...

故事小说类在很多公号的征稿信息中挺多的，我看了也很心动，什么千字500，千字1000，想着写一个字

赚一元钱真是爽呆了。但是写好一部故事小说也绝非易事。接下来就直接分享干货吧！①小说市场的现状

常书涵 (/u/563f5efb0a04?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

当你撑不下去的时候，请看看这18张照片 (/p/20d0c1... (/p/20d0c1a546ad?

01 在车上看到这样一幕，送外卖的小哥，背后背着一个婴儿。半旧的摩托
车，驮着生活，还载着未来，一定很重很重吧。生活到底有多难呢？有些人光

做个不俗人 (/u/c95721d10e6d?

utm_campaign=maleskine&utm_content=user&utm_medium=pc_all_hots&utm_source=recommendation)

一文读懂自然语言处理 (/p/416f7cc26ba2?utm_campaign=maleskine&u...

前言 自然语言处理是文本挖掘的研究领域之一，是人工智能和语言学领域的分支学科。在此领域中探讨如何
处理及运用自然语言。对于自然语言处理的发展历程，可以从哲学中的经验主义和理性主义说起。基于

笛在月明 (/u/f7e9d7a0e9ae?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

如何计算两个文档的相似度（二） (/p/303784312392?utm_campaign=...

【转自我爱自然语言】如何计算两个文档的相似度（二） | 我爱自然语言处理 二、gensim的安装和使用
1、安装gensim依赖NumPy和SciPy这两大Python科学计算工具包，一种简单的安装方法是pip install，但

邓旭东HIT (/u/1562c7f16a04?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/a9445f709e8e?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

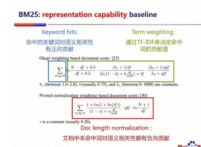
自然语言处理真实项目实战（20170830） (/p/a9445f709e8e?utm_cam...

前言 本文根据实际项目撰写，由于项目保密要求，源代码将进行一定程度的删减。本文撰写的目的是进行
公司培训，请勿以任何形式进行转载。由于是日语项目，用到的分词软件等，在中文任务中需要替换为相

中和软件技术推进 (/u/b19707134332?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

(/p/3a9f49834c4a?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

浅谈智能搜索和对话式OS (/p/3a9f49834c4a?utm_campaign=maleskin...

前面的文章主要从理论的角度介绍了自然语言人机对话系统所可能涉及到的多个领域的经典模型和基础知
识。这篇文章，甚至之后的文章，会更贴近业务的角度来写，侧重于介绍一些与自然语言问答业务密切

我偏笑_NS Nirvana (/u/2293f85dc197?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

关于这个美好的世界 (/p/5aaa163630b7?utm_campaign=maleskine&ut...

我所辅导的这个小朋友上小学二年级 他把他的名字用铅笔 小心翼翼写在新书上，并抬头看我“我喜欢钢笔
不喜欢铅笔” 这孩子草原上撒下黑色的雪 久久不能融化，也不让人瑟瑟发抖 那么明天到了学校。孩子们

羊在东 (/u/3d4454990c31?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)


(/p/79673d736eeb?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

央美原创教你素描静物绘画诀窍 (/p/79673d736eeb?utm_campaign=ma...

静物素描是基础中的基础 容易表现出效果 同时也容易暴露基本功的弊端 不少同学们感觉学了这么多年绘画 静物素描还是画不好 今天,央美原创 将给大家解读素描静物小诀窍 体现生活气息,合乎情理 这是首要

 央美原创美术培训中心 (/u/ecd60eef0d35?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)


(/p/bb72e4e9aa1d?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

纪念志摩去世四周年 (/p/bb72e4e9aa1d?utm_campaign=maleskine&ut...


在楼下的咖啡馆小坐,发现了林徽因的《你是人间的四月天》,一打开就是这篇文章,字里行间都流露了作者的感。每桩事都像是造物的意旨,归根都是运命,但我明知道每桩事都像有我们自己的影子在里面

 张漪纹 (/u/18d42f04df86?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)

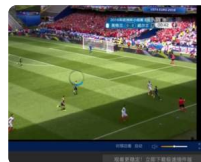
打造人格魅力的秘诀 (/p/29fb0b130308?utm_campaign=maleskine&ut...

如何拥有的人格魅力 什么是人格魅力? 何谓人格魅力? 首先要弄清什么是人格。人格是指人的性格、气质、能力等特征的总和,也指个人的道德品质和人的能作为权力、义务的主体的资格。而人格魅力则指一个人

 之乎者也06 (/u/a3b731ae2ed7?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)


(/p/94695d019e84?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendation)

移动体育直播如何达到广电直播流畅度 | 架构师实践日 (/p/94695d019e8...

编者按: 2016 年是直播元年,体育赛事直播发生了从广电垄断的 PGC 模式到 PGC + UGC 模式的转变。PGC 模式使用专有线路和卫星信号,其网络质量可以保证观众观看电视时无卡顿现象。而 PGC + UGC 模

 七牛云 (/u/342c4dafa482?

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendation)