自然语言处理之LDA主题模型

原创 大邓 大邓和他的Python 2018-11-30

话题模型

话题模型 是为发现文档集合中的 **话题** 而开发出来的一种统计方法。常见的话题模型有LSA、PLSA、LDA，其中LDA（Latent Dirichlet Allocation)是表现最好的话题模型。LDA也被称为三层贝叶斯概率模型，包含词语、话题和文档三层结构。我们认为一篇文章的产生是服从概率分布的，即每个词都是通过“以一定概率选择了某个话题，并从这个话题中以一定的概率选择了某个词语”。

LatentDirichletAllocation

在 sklearn.decomposition.LatentDirichletAllocation 提供了LDA的实现，LatentDirichletAllocation常用的需要设置的参数包括：

- n_topics: 文档集中隐藏的话题数目K。
- random_state: 随机状态码，保证每次程序运行都能得到相同的随机，使得每次程序之间可以比较运行效果。

LatentDirichletAllocation实际上还有很多参数可供选择，具体可以参考sklearn的文档学习，本文只提供基本参数的学习。

实验数据

下面我们准备了四段文本，分别存放在 `test1.txt`, `test2.txt`, `test3.txt` 和 `test4.txt` 。我们打开看看这四段文本的内容：

> **test1.txt**
> “红色联合”对“四·二八兵团”总部大楼的攻击已持续了两天，他们的旗帜在大楼周围躁动地飘扬着，仿佛渴望干柴的火种。“红色联合”的指挥官心急如焚，他并不惧怕大楼的守卫者，那二百多名“四·二八”战士，与诞生于l966年初、经历过大检阅和大串联的“红色联合”相比要稚嫩许多。他怕的是大楼中那十几个大铁炉子，里面塞满了烈性炸药，用电雷管串联起来，他看不到它们，但能感觉到它们磁石般的存在，开关一合，玉石俱焚，而“四·二八”的那些小红卫兵们是有这个精神力量的。比起已经在风雨中成熟了许多的第一代红卫兵，新生的造反派们像火炭上的狼群，除了疯狂还是疯狂。
> **test2.txt**

> 大楼顶上出现了一个娇小的身影，那个美丽的女孩子挥动着一面“四·二八”的大旗，她的出现立刻招来了一阵杂乱的枪声，射击的武器五花八门，有陈旧的美式卡宾枪、捷克式机枪和三八大盖，也有崭新的制式步枪和冲锋枪——后者是在“八月社论”发表之后从军队中偷抢来的（注：1967年8月《红旗》杂志发表“揪军内一小撮”的社论，使冲击军区、抢夺军队枪支弹药的事件愈演愈烈，全国范围的武斗也进入高·潮。）
>
> **test3.txt**
>
> 话说天下大势，分久必合，合久必分。周末七国分争，并入于秦。及秦灭之后，楚、汉分争，又并入于汉。汉朝自高祖斩白蛇而起义，一统天下，后来光武中兴，传至献帝，遂分为三国。推其致乱之由，殆始于桓、灵二帝。桓帝禁锢善类，崇信宦官。及桓帝崩，灵帝即位，大将军窦武、太傅陈蕃共相辅佐。时有宦官曹节等弄权，窦武、陈蕃谋诛之，机事不密，反为所害，中涓自此愈横。
>
> **test4.txt**
>
> 时巨鹿郡有兄弟三人，一名张角，一名张宝，一名张梁。那张角本是个不第秀才，因入山采药，遇一老人，碧眼童颜，手执藜杖，唤角至一洞中，以天书三卷授之，曰：“此名《太平要术》，汝得之，当代天宣化，普救世人；若萌异心，必获恶报。”角拜问姓名。老人曰：“吾乃南华老仙也。”言讫，化阵清风而去。角得此书，晓夜攻习，能呼风唤雨，号为“太平道人”。中平元年正月内，疫气流行，张角散施符水，为人治病，自称“大贤良师”。

LDA分析步骤：

1. 读取数据，并分词
2. 去除停用词
3. 构建Tfidf矩阵，每一行代表一个test的文档，每一列代表一个词语的tfidf值
4. LDA分析（fit和transform），输出结果

实验

1.读取数据与分词

实验数据存放在data文件夹中，test1.txt, test2.txt, test3.txt和test4.txt。

```
import jieba

filepaths = ['data/test1.txt', 'data/test2.txt', 'data/test3.txt', 'data/test4.txt']

docs = [open(f).read() for f in filepaths]

docs = [jieba.lcut(doc)
        for doc in docs]

docs
```

2.去除停止词

实际操作中需要自己构建停用词表，然后剔除掉这些无意义的词语，在本文中去停止词操作比较简单粗暴了点，只保留词语长度大于1的。

```
docs = [[w
         for w in doc
         if len(w)>1]
         for doc in docs]

#sklearn默认分析的语言是英文，我们要组织成类似英文那样以空格间隔的语言形式。
#corpus现在是一个列表，列表中有四个字符串。
#每个字符串就是一个文档
corpus = [' '.join(doc)
          for doc in docs]
corpus
```

运行

```
['红色 联合 二八 兵团 总部 大楼 攻击 持续 两天 他们 旗帜 大楼 周围 躁动 飘扬 仿佛 渴望 干柴 火
 '大楼 顶上 出现 一个 娇小 身影 那个 美丽 女孩子 挥动 一面 二八 大旗 出现 立刻 招来 一阵 杂乱
 '巨鹿郡 兄弟 三人 一名 张角 一名 张宝 一名 张梁 张角本 秀才 因入 采药 遇一 老人 碧眼 童颜 手执
 '天下 大势 分久必合 合久必分 周末 分争 并入 秦灭 之后 分争 并入 于汉 汉朝 高祖 白蛇 起义 一统
```

3. 构建Tfidf矩阵

每一行代表一个test的文档，每一列代表一个词语的tfidf值。学过的sklearn的都知道fit和transform的意义，如果对tfidf不懂的可以查看咱们之前分享的文章。

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import LatentDirichletAllocation


tfidf = TfidfVectorizer()
tfidf_matrix = tfidf.fit_transform(corpus)
tfidf_matrix
```

运行

```
<4x496 sparse matrix of type '<class 'numpy.float64'>'
    with 513 stored elements in Compressed Sparse Row format>
```

tfidf_matrix是 4x496 ，即4行496列，其中

- 4行指的是四个文档
- 496列是496个词语（也就是语料中一共出现了496个词语）

4. LDA分析（fit和transform）

同上，这里也分为fit和transform，由于我们有预先的知识，知道这四个文档来源于三国和三体，所以话题数K天然的等于2，即 `n_topics=2` 。

由于LDA属于聚类分析的一种，而聚类分析过程中会随机初始化，为了保证你也能得到与大邓一样的运行结果，我设置了random_state=123456。当然设置成别的数字也可以，这里的random_state相当于口令，咱们两个口令一致才能得到相同的答案。如果你换了random_state，那么咱们两个得到的结果可能会有出入。

```
lda = LatentDirichletAllocation(n_topics=2,
                                random_state=123456)
docres = lda.fit_transform(tfidf_matrix)
docres
```

运行

```
array([[0.91159844, 0.08840156],
       [0.93385048, 0.06614952],
       [0.06859599, 0.93140401],
       [0.06916256, 0.93083744]])
```

得到的结果是 `4*2` 的矩阵。行表示文档，列表示话题。我们将第一列认定为话题1，第二列认定为话题2

```
test1.txt 对应着[0.91159844, 0.08840156], 属于话题1
test2.txt 对应着[0.93385048, 0.06614952], 属于话题1
test3.txt 对应着[0.06859599, 0.93140401], 属于话题2
test4.txt 对应着[0.06916256, 0.93083744], 属于话题2
```

矩阵中的值是隶属于某一话题的概率，比如test1.txt隶属于话题1的概率为0.91159844，隶属于话题2的概率为0.08840156，所以我们选择最大的值所属的概率。最终判断test1.txt是话题1.

大家都明白文言文与现代文区别是很大的，test1.txt和test2.txt是现代文，而test3.txt和test4.txt是文言文，而sklearn也将这四个文本准确的分为两类。

长按小程序码即可获取本文的jupyter notebeook下载链接