

法律声明

□ 本课件包括演示文稿、示例、代码、题库、视频和声音等内容，小象学院和主讲老师拥有完全知识产权的权利；只限于善意学习者在本课程使用，不得在课程范围外向任何第三方散播。任何其他人或机构不得盗版、复制、仿造其中的创意及内容，我们保留一切通过法律手段追究违反者的权利。

□ 课程详情请咨询

■ 微信公众号：小象

■ 新浪微博：ChinaHadoop



主题模型



小象学院
ChinaHadoop.cn

邹博

主要内容

- LDA开源实现库介绍
- LDA1.0.4/Gensim的使用
- 复习：
 - TF-IDF
 - 相似度计算

Crawler爬取数据

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	区域	小区名称	户型	面积	价格(万)	单价(元/平米)	性质	朝向	装修	是否有电梯	楼层	建筑年代	楼型
2	东城	桃杨路北里12号院	3室1厅	98.17	638	64990	房本满五年	东西	简装	无电梯	中楼层(共6层)	1999	板楼
3	东城	民安小区东扬威街	3室1厅	103.29	1000	96815	房本满五年	西南北	简装	有电梯	低楼层(共16层)	2002	塔楼
4	东城	民安小区东直门内北小街	2室1厅	74.54	732	98203	房本满五年	南北	简装	有电梯	高楼层(共16层)	2003	板楼
5	东城	海运仓小区	2室1厅	64.82	700	107992	房本满五年	南北	精装	无电梯	中楼层(共7层)	2003	板楼
6	东城	新景家园西区	1室1厅	57.18	560	97937	房本满五年	西南	精装	有电梯	中楼层(共17层)	2004	板塔结合
7	东城	小羊宜宾胡同	4室1厅	93.81	910	97005	房本满五年	南北	简装	有电梯	低楼层(共16层)	1992	塔楼
8	东城	禾风相府	3室2厅	163.83	1801	109932	房本满五年	南北	其他	有电梯	低楼层(共14层)	2008	板塔结合
9	东城	西革新里110号院	2室1厅	61.4	400	65147	房本满五年	南北	简装	无电梯	高楼层(共6层)	1992	板楼
10	东城	东中街	4室1厅	200.77	1770	88161	房本满五年	南北	精装	有电梯	中楼层(共9层)	1999	板塔结合
11	东城	东花市北里中区	3室2厅	98.65	990	100355	房本满五年	南北	精装	无电梯	中楼层(共5层)	1994	板楼
12	东城	民安小区东直门北中街	2室1厅	75.11	722	96126	房本满五年	东西	精装	有电梯	中楼层(共11层)	2002	板塔结合
13	东城	中景濠庭	1室1厅	74.21	630	84895	房本满五年	北	精装	有电梯	低楼层(共22层)	2001	板塔结合
14	东城	华府景园	3室2厅	149.97	1350	90019	房本满五年	东西	简装	有电梯	中楼层(共12层)	2001	板塔结合
15	东城	朝阳门南小街	2室1厅	69.39	650	93674	房本满五年	东西	简装	有电梯	低楼层(共8层)	2004	板塔结合
16	东城	兴隆都市馨园	3室2厅	130.7	980	74981	房本满五年	南北	精装	无电梯	高楼层(共6层)	2003	板塔结合
17	东城	保利蔷薇	2室2厅	88.63	960	108316	房本满五年	西北	精装	有电梯	中楼层(共16层)	2008	板塔结合
18	东城	小羊宜宾胡同	2室1厅	61.77	650	105230	房本满五年	南	简装	有电梯	低楼层(共16层)	1992	塔楼
19	东城	安化北里	3室1厅	80.34	730	90864	房本满五年	西南	简装	有电梯	中楼层(共17层)	1989	塔楼
20	东城	香饵胡同	2室1厅	67.92	815	119995	房本满五年	南北	精装	无电梯	低楼层(共6层)	2002	板楼
21	东城	什锦花园胡同21号院	2室1厅	48	650	135417	房本满五年	南北	精装	无电梯	高楼层(共5层)	1985	板楼
22	东城	富贵园一区	3室2厅	144.45	1650	114227	房本满五年	南北	精装	有电梯	低楼层(共11层)	2003	板楼
23	东城	富莱茵花园	3室1厅	98.38	570	57939	房本满五年	南西北	简装	有电梯	中楼层(共21层)	1995	塔楼
24	东城	丽水湾畔家园	2室1厅	130	1100	84616	房本满五年	西	简装	有电梯	高楼层(共21层)	2002	塔楼
25	东城	国瑞城中区	3室1厅	90.47	880	97270	房本满五年	西南	精装	有电梯	低楼层(共16层)	2006	板塔结合
26	东城	金鱼池中区	4室1厅	139.34	850	61002	房本满五年	南北	精装	无电梯	低楼层(共6层)	2002	板楼
27	东城	新景家园东区	3室1厅	92.82	1030	110968	房本满五年	南	精装	有电梯	高楼层(共13层)	2002	板塔结合
28	东城	本家润园三期	3室1厅	97.45	930	95434	房本满五年	南西	精装	有电梯	高楼层(共18层)	2005	板塔结合
29	东城	新景家园东区	1室1厅	59.68	595	99699	房本满五年	南	简装	有电梯	中楼层(共17层)	2002	板楼
30	东城	北京上舍	1室0厅	61.2	580	94772	房本满五年	北	简装	有电梯	高楼层(共15层)	2008	塔楼
31	东城	和平里七区	2室1厅	41.82	450	107605	房本满五年	南	简装	无电梯	高楼层(共5层)	1983	板楼
32	东城	东土城路13号院	1室0厅	51.15	372	72728	房本满两年	南	精装	无电梯	中楼层(共5层)	1994	板楼
33	东城	金世纪嘉园	2室1厅	116.89	950	81273	房本满五年	东南	精装	有电梯	高楼层(共21层)	2003	塔楼
34	东城	民安小区民安街	2室1厅	74.29	868	116840	房本满五年	南北	精装	无	低楼层(共6层)	2002	板楼
35	东城	金鱼池中区	4室2厅	135.91	950	69900	房本满五年	南北	简装	无电梯	高楼层(共6层)	2002	板塔结合
36	东城	海晟名苑	2室1厅	140.53	1550	110297	房本满五年	东西	简装	有电梯	低楼层(共20层)	2003	板塔结合
37	东城	京城仁合	2室2厅	114.51	1010	88202	房本满五年	西南北	精装	有电梯	高楼层(共16层)	2002	板塔结合
38	东城	西水井胡同	2室1厅	74.95	680	90728	房本满五年	东西	精装	有电梯	中楼层(共16层)	2003	板塔结合
39	东城	新奥洋房	2室1厅	93.76	698	74446	房本满五年	东	精装	有电梯	中楼层(共11层)	2005	板塔结合
40	东城	西水井胡同	2室1厅	74.95	690	92062	房本满五年	东西	简装	有电梯	中楼层(共16层)	2003	板塔结合
41	东城	仓南胡同	2室1厅	77.8	920	118252	房本满五年	东西	简装	无电梯	中楼层(共6层)	2003	板楼
42	东城	新奥洋房	2室1厅	123.21	910	73858	房本满五年	东南西	精装	有电梯	高楼层(共13层)	2006	板塔结合
43	东城	北河沿大街	1室1厅	41.47	490	118158	房本满两年	南	简装	无电梯	高楼层(共6层)	1987	板楼
44	东城	东直门内大街	2室2厅	80.52	880	109290	房本满五年	东南	精装	有电梯	高楼层(共16层)	2003	塔楼

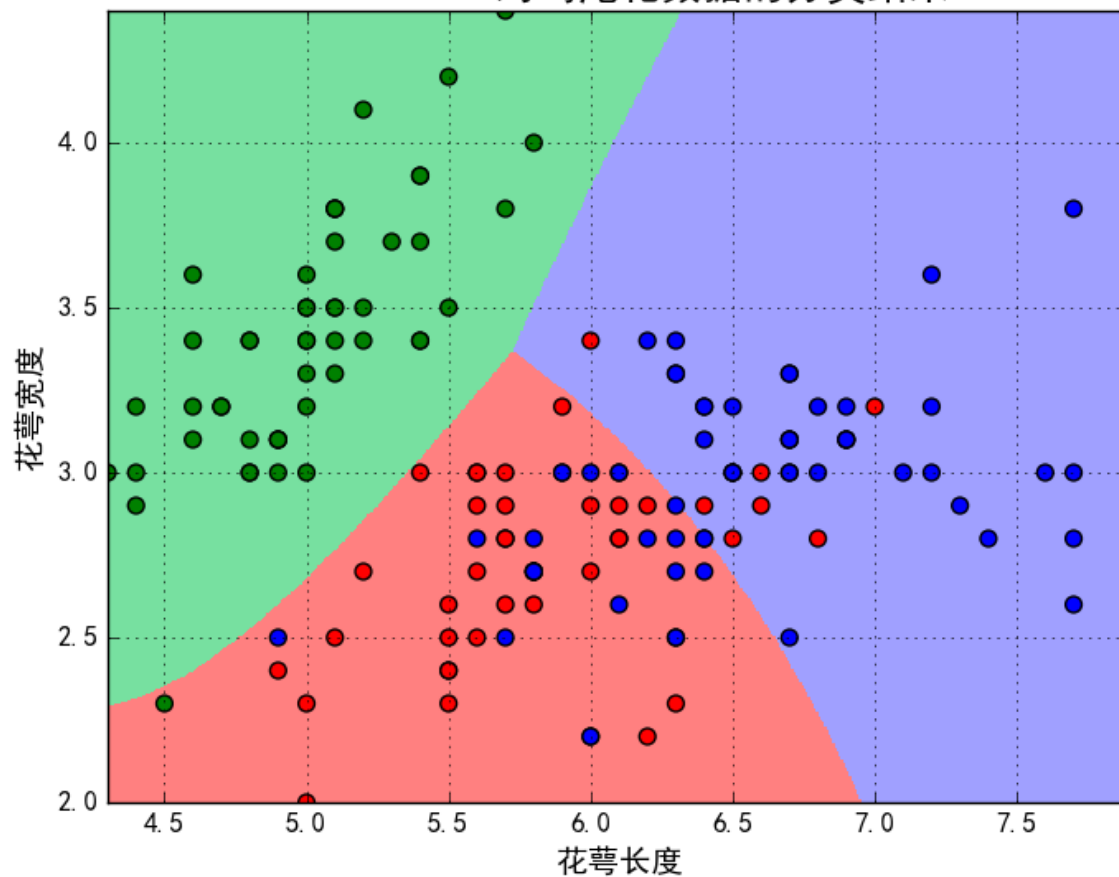
Code

```
writer = csv.writer(f)
writer.writerow(['区域', '小区名称', '户型', '面积', '价格(万)', '单价(元/平米)',
                '性质', '朝向', '装修', '是否有电梯', '楼层', '建筑年代', '楼型'])
res = requests.get('http://bj.lianjia.com/ershoufang')
res = res.text.encode(res.encoding).decode('utf-8')
soup = BeautifulSoup(res, 'html.parser')
# print soup.prettify()
districts = soup.find(name='div', attrs={'data-role': 'ershoufang'}) # <div data-role="ers
for district in districts.find_all(name='a'):
    print district['title']
    district_name = district.text # '东城', '西城', '朝阳', '海淀'.....
    url = '%s%s' % (url_main, district['href'])
    print url
    res = requests.get(url)
    res = res.text.encode(res.encoding).decode('utf-8')
    soup = BeautifulSoup(res, 'html.parser')
    # print soup.prettify()
    page = soup.find('div', {'class': 'page-box house-1st-page-box'})
    if not page: # 平谷区没有房源, 直接返回
        continue
    total_pages = dict(eval(page['page-data']))['totalPage'] # 总页数
    # print total_pages
    for j in range(1, total_pages+1):
        url_page = '%spg%d/' % (url, j)
        print 'url_page = ', url_page
        res = requests.get(url_page)
        res = res.text.encode(res.encoding).decode('utf-8')
        soup = BeautifulSoup(res, 'html.parser')
        # print soup.prettify()
        sells = soup.find(name='ul', attrs={'class': 'sellListContent', 'log-mod': 'list'})
```

GaussianNB



GaussianNB对鸢尾花数据的分类结果



GaussianNB / MultinomialNB

```
np.random.seed(0)
M = 20
N = 5
x = np.random.randint(2, size=(M, N))      # [low, high)
x = np.array(list(set([tuple(t) for t in x])))
M = len(x)
y = np.arange(M)
print '样本个数: %d, 特征数目: %d' % x.shape
print '样本: \n', x
mnf = MultinomialNB(alpha=1)      # 动手: 换成GaussianNB(
mnf.fit(x, y)
y_hat = mnf.predict(x)
print '预测类别: ', y_hat
print '准确率: %.2f%%' % (100*np.mean(y_hat == y))
print '系统得分: ', mnf.score(x, y)
```

20.1.Iris_GaussianNB

20.2.MultinomialNB_intro

20.3.text_classification

[0 0 0 0 1]

[1 0 0 1 0]

[1 1 1 1 1]

[0 1 1 1 1]

[1 1 0 0 0]

预测类别: [0 1 0 3 4 5 6 7 8 9 10 11 12 13 2 15 16]

准确率: 88.24%

系统得分: 0.882352941176

2 : [0 0 0 0 0] 被认为与 [1 1 0 1 0] 一个类别

14 : [1 1 1 1 1] 被认为与 [0 0 0 0 0] 一个类别

朴素贝叶斯的假设

- 一个特征出现的概率，与其他特征(条件)独立(特征独立性)
 - 其实是：对于给定分类的条件下，特征独立
- 每个特征同等重要(特征均衡性)

朴素贝叶斯的推导

- 朴素贝叶斯(Naive Bayes, NB)是基于“特征之间是独立的”这一朴素假设，应用贝叶斯定理的监督学习算法。
- 对于给定的特征向量 x_1, x_2, \dots, x_n
- 类别 y 的概率可以根据贝叶斯公式得到：

$$P(y | x_1, x_2, \dots, x_n) = \frac{P(y)P(x_1, x_2, \dots, x_n | y)}{P(x_1, x_2, \dots, x_n)}$$

朴素贝叶斯的推导

□ 使用朴素的独立性假设：

$$P(x_i | y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y)$$

□ 类别 y 的概率可简化为：

$$P(y | x_1, x_2, \dots, x_n) = \frac{P(y)P(x_1, x_2, \dots, x_n | y)}{P(x_1, x_2, \dots, x_n)} = \frac{P(y) \prod_{i=1}^n P(x_i | y)}{P(x_1, x_2, \dots, x_n)}$$

□ 在给定样本的前提下， $P(x_1, x_2, \dots, x_n)$ 是常数：

$$P(y | x_1, x_2, \dots, x_n) \propto P(y) \prod_{i=1}^n P(x_i | y)$$

□ 从而：
$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y)$$

高斯朴素贝叶斯 Gaussian Naive Bayes

- 根据样本使用MAP(Maximum A Posteriori)估计 y ，建立合理的模型估计 $P(x_i | y)$ ，从而得到样本的类别。

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y)$$

- 假设特征服从高斯分布，即：

$$P(x_i | y) = \frac{1}{\sqrt{2\pi}\sigma_y} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

- 参数使用MLE估计即可。

多项分布朴素贝叶斯Multinomial Naive Bayes

- 假设特征服从多项分布，从而，对于每个类别 y ，参数为 $\theta_y = (\theta_{y1}, \theta_{y2}, \dots, \theta_{yn})$ ，其中 n 为特征的数目， $P(x_i | y)$ 的概率为 θ_{yi} 。
- 参数 θ_y 使用MLE估计的结果为： $\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha \cdot n}$ ， $\alpha \geq 0$
- 假定训练集为 T ，有：
$$\begin{cases} N_{yi} = \sum_{x \in T} x_i \\ N_y = \sum_{i=1}^{|T|} N_{yi} \end{cases}$$
- 其中，
 - $\alpha = 1$ 称为Laplace平滑，
 - $\alpha < 1$ 称为Lidstone平滑。

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i | y)$$

以文本分类为例

- 样本：1000封邮件，每个邮件被标记为垃圾邮件或者非垃圾邮件
- 分类目标：给定第1001封邮件，确定它是垃圾邮件还是非垃圾邮件
- 方法：朴素贝叶斯

分析

- 类别C: 垃圾邮件 C_1 , 非垃圾邮件 C_2
- 词汇表, 两种建立方法:
 - 使用现成的单词词典;
 - 将所有邮件中出现的单词都统计出来, 得到词典。
 - 记单词数目为N
- 将每个邮件m映射成维度为N的向量x
 - 若单词 w_i 在邮件m中出现过, 则 $x_i=1$, 否则, $x_i=0$ 。即邮件的向量化: $m \rightarrow (x_1, x_2, \dots, x_N)$
- 贝叶斯公式: $P(c|x) = P(x|c) * P(c) / P(x)$
 - $P(c_1|x) = P(x|c_1) * P(c_1) / P(x)$
 - $P(c_2|x) = P(x|c_2) * P(c_2) / P(x)$
 - 注意这里x是向量

分解

- $P(c|x) = P(x|c) * P(c) / P(x)$
- $P(x|c) = P(x_1, x_2 \dots x_N | c) = P(x_1|c) * P(x_2|c) \dots P(x_N|c)$
 - 特征条件独立假设
- $P(x) = P(x_1, x_2 \dots x_N) = P(x_1) * P(x_2) \dots P(x_N)$
 - 特征独立假设
- 带入公式: $P(c|x) = P(x|c) * P(c) / P(x)$

- 等式右侧各项的含义:
 - $P(x_i|c_j)$: 在 c_j (此题目, c_j 要么为垃圾邮件1, 要么为非垃圾邮件2) 的前提下, 第 i 个单词 x_i 出现的概率
 - $P(x_i)$: 在所有样本中, 单词 x_i 出现的概率
 - $P(c_j)$: 在所有样本中, 邮件类别 c_j 出现的概率

拉普拉斯平滑

- $p(x_1|c_1)$ 是指的:在垃圾邮件 c_1 这个类别中, 单词 x_1 出现的概率。
 - x_1 是待考察的邮件中的某个单词
- 定义符号
 - n_1 : 在所有垃圾邮件中单词 x_1 出现的次数。如果 x_1 没有出现过, 则 $n_1=0$ 。
 - n : 属于 c_1 类的所有文档的出现过的单词总数目。
- 得到公式:
$$p(x_1|c_1) = \frac{n_1}{n}$$
- 拉普拉斯平滑:
$$p(x_1|c_1) = \frac{n_1 + 1}{n + N}$$
 - 其中, N 是所有单词的数目。修正分母是为了保证概率和为1
- 同理, 以同样的平滑方案处理 $p(x_1)$

对朴素贝叶斯的思考

- 拉普拉斯平滑能够避免0/0带来的算法异常
- 要比较的是 $P(c1|x)$ 和 $P(c2|x)$ 的相对大小，而根据公式 $P(c|x) = P(x|c) \cdot P(c) / P(x)$ ，二者的分母都是除以 $P(x)$ ，实践时可以不计算该系数。
- 编程的限制：小数乘积下溢出怎么办？
- 问题：一个词在样本中出现多次，和一个词在样本中出现一次，形成的词向量相同
 - 由0/1向量改成频数向量或TF-IDF向量
- 如何判断两个文档的距离
 - 夹角余弦
- 如何给定合适的超参数 $\hat{\theta}_{yi} = \frac{N_{yi} + \alpha}{N_y + \alpha \cdot n}$, $\alpha \geq 0$
 - 交叉验证

LDA的实现

- LDA-C: David Blei, C实现, VBEM参数估计
 - <http://www.cs.princeton.edu/~blei/lda-c/index.html>
- GibbsLDA++/JGibbLDA : C/C++实现/Java实现
 - <http://gibbslda.sourceforge.net/> <http://jgibblda.sourceforge.net/>
 - Xuan-Hieu Phan/Cam-Tu Nguyen, 输入输出一致
- Matlab Topic Modeling Toolbox 1.4, Mark Steyvers, Gibbs采样
 - http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm
- Gensim: Online VB
 - 官网: <http://radimrehurek.com/gensim/index.html>
 - github: http://www.cs.columbia.edu/~blei/topicmodeling_software.html
- Scikit-learn: sklearn.decomposition.LatentDirichletAllocation/Online VB
- LDA/Online VB: <https://pypi.python.org/pypi/lda>
- LDA不完全列表:
 - http://www.cs.columbia.edu/~blei/topicmodeling_software.html

Country	Year	Value
Algeria	2010	0.00
Algeria	2011	0.00
Algeria	2012	0.00
Algeria	2013	0.00
Algeria	2014	0.00
Algeria	2015	0.00
Algeria	2016	0.00
Algeria	2017	0.00
Algeria	2018	0.00
Algeria	2019	0.00
Algeria	2020	0.00
Algeria	2021	0.00
Algeria	2022	0.00
Algeria	2023	0.00
Algeria	2024	0.00
Algeria	2025	0.00
Algeria	2026	0.00
Algeria	2027	0.00
Algeria	2028	0.00
Algeria	2029	0.00
Algeria	2030	0.00
Algeria	2031	0.00
Algeria	2032	0.00
Algeria	2033	0.00
Algeria	2034	0.00
Algeria	2035	0.00
Algeria	2036	0.00
Algeria	2037	0.00
Algeria	2038	0.00
Algeria	2039	0.00
Algeria	2040	0.00
Algeria	2041	0.00
Algeria	2042	0.00
Algeria	2043	0.00
Algeria	2044	0.00
Algeria	2045	0.00
Algeria	2046	0.00
Algeria	2047	0.00
Algeria	2048	0.00
Algeria	2049	0.00
Algeria	2050	0.00
Algeria	2051	0.00
Algeria	2052	0.00
Algeria	2053	0.00
Algeria	2054	0.00
Algeria	2055	0.00
Algeria	2056	0.00
Algeria	2057	0.00
Algeria	2058	0.00
Algeria	2059	0.00
Algeria	2060	0.00
Algeria	2061	0.00
Algeria	2062	0.00
Algeria	2063	0.00
Algeria	2064	0.00
Algeria	2065	0.00
Algeria	2066	0.00
Algeria	2067	0.00
Algeria	2068	0.00
Algeria	2069	0.00
Algeria	2070	0.00
Algeria	2071	0.00
Algeria	2072	0.00
Algeria	2073	0.00
Algeria	2074	0.00
Algeria	2075	0.00
Algeria	2076	0.00
Algeria	2077	0.00
Algeria	2078	0.00
Algeria	2079	0.00
Algeria	2080	0.00
Algeria	2081	0.00
Algeria	2082	0.00
Algeria	2083	0.00
Algeria	2084	0.00
Algeria	2085	0.00
Algeria	2086	0.00
Algeria	2087	0.00
Algeria	2088	0.00
Algeria	2089	0.00
Algeria	2090	0.00
Algeria	2091	0.00
Algeria	2092	0.00
Algeria	2093	0.00
Algeria	2094	0.00
Algeria	2095	0.00
Algeria	2096	0.00
Algeria	2097	0.00
Algeria	2098	0.00
Algeria	2099	0.00
Algeria	2100	0.00
Algeria	2101	0.00
Algeria	2102	0.00
Algeria	2103	0.00
Algeria	2104	0.00
Algeria	2105	0.00
Algeria	2106	0.00
Algeria	2107	0.00
Algeria	2108	0.00
Algeria	2109	0.00
Algeria	2110	0.00
Algeria	2111	0.00
Algeria	2112	0.00
Algeria	2113	0.00
Algeria	2114	0.00
Algeria	2115	0.00
Algeria	2116	0.00
Algeria	2117	0.00
Algeria	2118	0.00
Algeria	2119	0.00
Algeria	2120	0.00
Algeria	2121	0.00
Algeria	2122	

Collecting gensim

100%  4.2MB 141kB/s

Requirement already satisfied (use --upgrade to upgrade): six>=1.5.0 in c:\python27\lib\site-packages (from gensim)

Collecting smart-open>=1.2.1 (from gensim)

Collecting boto>=2.32 (from smart-open>=1.2.1->gensim)

100% 1.4MB 249kB/s

```

...
Downloading bz2file-0.98.tar.gz

```

```
Downloading requests-2.11.1-py2.py3-none-any.whl (514kB)
```

100% 522kB 333kB/s

```
Installing collected packages: boto, bz2file, requests, smart-open, gensim
```

Running setup.py install for bz2file ... done

Running setup.py install for smart-open ... done

Successfully installed boto-2.42.0 bz2file-0.98 gensim-0.13.2 requests-2.11.1 smart-open-1.3.4

TF-IDF

Text =

```
[['human', 'machine', 'interface', 'lab', 'abc', 'computer', 'applications'],  
 ['survey', 'user', 'opinion', 'computer', 'system', 'response', 'time'],  
 ['eps', 'user', 'interface', 'management', 'system'],  
 ['system', 'human', 'system', 'engineering', 'testing', 'eps'],  
 ['relation', 'user', 'perceived', 'response', 'time', 'error', 'measurement'],  
 ['generation', 'random', 'binary', 'unordered', 'trees'],  
 ['intersection', 'graph', 'paths', 'trees'],  
 ['graph', 'minors', 'iv', 'widths', 'trees', 'well', 'quasi', 'ordering'],  
 ['graph', 'minors', 'survey']]
```

TF-IDF:

```
[(0, 0.4301019571350565), (1, 0.4301019571350565), (2, 0.4301019571350565), (3, 0.4301019571350565), (4, 0.2944198962221451), (5  
[(4, 0.3726494271826947), (7, 0.27219160459794917), (8, 0.3726494271826947), (9, 0.27219160459794917), (10, 0.3726494271826947),  
[(6, 0.438482464916089), (7, 0.32027755044706185), (9, 0.32027755044706185), (13, 0.6405551008941237), (14, 0.438482464916089)]  
[(5, 0.3449874408519962), (7, 0.5039733231394895), (14, 0.3449874408519962), (15, 0.5039733231394895), (16, 0.5039733231394895)]  
[(9, 0.21953536176370683), (10, 0.30055933182961736), (12, 0.30055933182961736), (17, 0.43907072352741366), (18, 0.4390707235274  
[(21, 0.48507125007266594), (22, 0.48507125007266594), (23, 0.48507125007266594), (24, 0.48507125007266594), (25, 0.242535625036  
[(25, 0.31622776601683794), (26, 0.31622776601683794), (27, 0.6324555320336759), (28, 0.6324555320336759)]  
[(25, 0.20466057569885868), (26, 0.20466057569885868), (29, 0.2801947048062438), (30, 0.40932115139771735), (31, 0.4093211513977  
[(8, 0.6282580468670046), (26, 0.45889394536615247), (29, 0.6282580468670046)]
```

LSI

LSI Model:

```
[[ (0, 0.34057117986841989), (1, -0.20602251622679696)],  
  [(0, 0.69330400021715577), (1, 0.0072327583903918488)],  
  [(0, 0.59026076703897357), (1, -0.35260469490855789)],  
  [(0, 0.52149018218251453), (1, -0.33887976154055377)],  
  [(0, 0.39533193176354431), (1, -0.059192853366596486)],  
  [(0, 0.036353173528493307), (1, 0.18146550208818862)],  
  [(0, 0.14709012328778862), (1, 0.49432948127822229)],  
  [(0, 0.21407117317565286), (1, 0.640645666445394)],  
  [(0, 0.40066568318170664), (1, 0.64131082990940158)]]
```

LSI Topics:

```
[(0,  
  u' 0.400*"system" + 0.318*"survey" + 0.290*"user" + 0.274*"eps" + 0.236*"management"',  
  (1,  
    u' 0.421*"minors" + 0.420*"graph" + 0.293*"survey" + 0.239*"trees" + 0.226*"intersection"')]
```

思考

- LSI/LFM/ICA的关系
- LSI和pLSA的关系

相似度

Similarity:

```
[array([ 1.          ,  0.85017949,  0.99998462,  0.99948108,  0.92283762,
        -0.33944285, -0.2520774 , -0.21974573,  0.01438823], dtype=float32),
 array([ 0.85017949,  1.          ,  0.85309052,  0.83277911,  0.98737705,
        0.20664607,  0.29518002,  0.32680073,  0.53867108], dtype=float32),
 array([ 0.99998462,  0.85309052,  1.          ,  0.99928677,  0.92496276,
        -0.33421332, -0.24669874, -0.214324  ,  0.01994151], dtype=float32),
 array([ 0.99948108,  0.83277911,  0.99928677,  1.          ,  0.90995121,
        -0.36956567, -0.28311783, -0.25105584, -0.01782739], dtype=float32),
 array([ 0.92283762,  0.98737705,  0.92496276,  0.90995121,  1.          ,
        0.04906873,  0.14012395,  0.1729846 ,  0.39842743], dtype=float32),
 array([-0.33944285,  0.20664607, -0.33421332, -0.36956567,  0.04906873,
         1.          ,  0.99581695,  0.99222624,  0.93564534], dtype=float32),
 array([-0.2520774 ,  0.29518002, -0.24669874, -0.28311783,  0.14012395,
         0.99581695,  1.          ,  0.99944651,  0.96397996], dtype=float32),
 array([-0.21974573,  0.32680073, -0.214324  , -0.25105584,  0.1729846 ,
         0.99222624,  0.99944651,  0.99999994,  0.97229445], dtype=float32),
 array([ 0.01438823,  0.53867108,  0.01994151, -0.01782739,  0.39842743,
         0.93564534,  0.96397996,  0.97229445,  1.          ], dtype=float32)]
```

主题和主题分布

LDA Model:

Document-Topic:

```
[[ (0, 0.68548441915170544), (1, 0.31451558084829462)],  
  [(0, 0.65732202058761513), (1, 0.34267797941238493)],  
  [(0, 0.67101883898793013), (1, 0.32898116101206987)],  
  [(0, 0.29774557750241137), (1, 0.70225442249758874)],  
  [(0, 0.55150516193766697), (1, 0.44849483806233303)],  
  [(0, 0.25456933670287446), (1, 0.7454306632971256)],  
  [(0, 0.67476418767307922), (1, 0.32523581232692073)],  
  [(0, 0.29509659300584296), (1, 0.7049034069941571)],  
  [(0, 0.69445879658152987), (1, 0.30554120341847024)]]
```

Topic 0

```
[(u' survey', 0.042573497130974247),  
 (u' minors', 0.03943557671036535),  
 (u' graph', 0.038776707760135178),  
 (u' system', 0.034575198665359616),  
 (u' trees', 0.032742027152788719),  
 (u' opinion', 0.031680224783503845),  
 (u' generation', 0.031141365123546434),  
 (u' unordered', 0.030981049002428096),  
 (u' time', 0.030911535753312992),  
 (u' random', 0.03090147631201922)]
```

Topic 1

```
[(u' system', 0.037724259436260198),  
 (u' eps', 0.03524885080697393),  
 (u' interface', 0.034303635122775261),  
 (u' intersection', 0.03398428810730824),  
 (u' user', 0.033982740385072041),  
 (u' management', 0.033477115230294417),  
 (u' human', 0.032957111835837112),  
 (u' paths', 0.032333361709319365),  
 (u' engineering', 0.030715385582341159),  
 (u' computer', 0.030706245324429286)]
```


LDA计算的相似度

Similarity:

```
[array([ 0.99999994,  0.79683411,  0.99871153,  0.9988395 ,  0.99509394,
         0.68600154,  0.98457313,  0.66293609,  0.71091771], dtype=float32),
 array([ 0.79683411,  1.          ,  0.82646847,  0.82500935,  0.85270071,
         0.98624408,  0.89025998,  0.98059875,  0.99140108], dtype=float32),
 array([ 0.99871153,  0.82646847,  1.          ,  0.99999666,  0.9988324 ,
         0.72204095,  0.99218392,  0.70007479,  0.74569058], dtype=float32),
 array([ 0.9988395 ,  0.82500935,  0.99999666,  1.          ,  0.99870408,
         0.72024882,  0.99185783,  0.69822526,  0.74396449], dtype=float32),
 array([ 0.99509394,  0.85270071,  0.9988324 ,  0.99870408,  0.99999994,
         0.75462055,  0.99705368,  0.7337535 ,  0.77700794], dtype=float32),
 array([ 0.68600154,  0.98624408,  0.72204095,  0.72024882,  0.75462055,
         1.          ,  0.80272919,  0.9995119 ,  0.9993937 ], dtype=float32),
 array([ 0.98457313,  0.89025998,  0.99218392,  0.99185783,  0.99705368,
         0.80272919,  1.          ,  0.78370738,  0.82300484], dtype=float32),
 array([ 0.66293609,  0.98059875,  0.70007479,  0.69822526,  0.7337535 ,
         0.9995119 ,  0.78370738,  1.          ,  0.99781829], dtype=float32),
 array([ 0.71091771,  0.99140108,  0.74569058,  0.74396449,  0.77700794,
         0.9993937 ,  0.82300484,  0.99781829,  1.          ], dtype=float32)]
```

网易新闻语料

16.news.dat - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

原 标题 猫咪荣誉站长去世 日本会津铁道办葬礼 中新网日电 据媒报道在日本福岛县会津若松市的会津铁道芦之牧温泉站从起一直担任该站荣誉站长的母猫 Bus 以推测年龄的高龄于近日离开了人世为此会津铁道以公司葬礼的形式为猫咪站长举行了葬礼 据报道在葬礼上约有人专程从县内外赶来参加到不得不站在车站楼外的人们在冰冷的雨中撑着伞祈祷保佑猫咪的地下之灵会津铁道社长兼葬礼委员会会长大石直在致辞中表示对于生性自由的 Bus 来说就任荣誉站长之后就一直被人群包围着这对她来说也许是痛苦但她那不撒娇献媚不喜欢就是不喜欢的率真个性我却欣赏在随后的致辞中县及市等相关人士还谈到了 Bus 的功劳称赞她提高了会津铁道与芦之牧温泉的知名度并为福岛招揽来了游客此外曾在年前拍摄制作 Bus 写真集香川县高松市的铁道摄影师坪内政美谈到我听说 Bus 是在送走了最后列车后才离开人世的一直到生命的最后她都秉承着铁路员工的精神看着人们将车站楼团团围住的身影作为葬主的站长小林美智子说道我深深感受到了大家对 Bus 的喜爱据了解 Bus 的遗体被葬在了铁轨旁的花桃树下读书有意思是一般的阅读另则是特指阅读书籍电子时代更需要强调的也许是后意思的读书因为阅读书籍读书比阅读电子屏幕文字读屏更是专注的阅读

原 标题 低价鱼精蛋白缺货 心脏病 人排队等药救心 业内人士分析 此次全国性缺货或是药企涨价的前兆 这个赵碧珍觉得特别漫长她患有心脏病需要开胸更换心脏瓣膜因为手术必用药鱼精蛋白缺货她只能在病房里排队等药她的病友们等不及已陆续离开她一直在等待但不知何时能等来救心的药鱼精蛋白全国性缺货今年并非年前也曾经出现过低价救命药越来越高频率出现缺货有医生分析这与药品价格低企业利润薄无生产积极性有关甚至有医生猜测此次可能是药品生产企业涨价的前兆无奈的等待住院等药救心 华西都市报记者能不能帮我买到鱼精蛋白药近日赵碧珍在走投无路的情况下向本报求助赵碧珍今年自贡人去年体检中她被查出患有风湿性心脏病心脏瓣膜出了问题需要更换需要修补她入住成都军区总医院心脏外科等待手术医生告诉她因为缺少名为鱼精蛋白的药手术没法进行我好不容易排到了床位不愿意轻易放弃赵碧珍说她总以为药应该缺不了说不准过就等来了药品于是她继续在医院住了下去过了药还是没来她的心情也越来越差据记者了解像赵碧珍一样等药做手术的病人不在少数的高含清的 心脏手术也是没药等待了小手手术被取消了在此期间其家人还在四川大学华西医院和省人民医院去打听了希望能借到药然而各大医院都说没药可借全国性药荒今年来已成全国性缺货成都军区总医院心血管外科主任张近宝说鱼精蛋白是心脏病 人做体外循环手术时必需用的药品根据体重

LDA

初始化停止词列表 --
开始读入语料数据 --
读入语料数据完成，用时9.256秒
文本数目：2043个
正在建立词典 --
正在计算文本向量 --
正在计算文档TF-IDF --
建立文档TF-IDF完成，用时0.185秒
LDA模型拟合推断 --
LDA模型完成，训练时间为 37.687秒

10个文档的主题分布：

第532个文档的前10个主题： [20 18 25 4 13 6 19 28 22 24]
[0.50757285 0.10239849 0.07296044 0.05082813 0.02763301 0.02729199
0.02345142 0.02105525 0.01749429 0.01665937]
第1043个文档的前10个主题： [23 14 4 18 10 24 6 15 0 20]
[0.4981378 0.06441008 0.05225744 0.05120348 0.0392068 0.03119684
0.02928527 0.02618645 0.02403664 0.02328459]
第1035个文档的前10个主题： [19 25 4 11 15 0 16 28 6 7]
[0.26742334 0.16533452 0.08484096 0.07141483 0.0688248 0.05389866
0.05103031 0.03916642 0.03554837 0.027476]
第588个文档的前10个主题： [7 20 5 12 19 21 15 17 23 14]
[0.26408634 0.20762942 0.1160332 0.10415797 0.06068137 0.05660975
0.02997539 0.01992539 0.01928632 0.01816658]
第1412个文档的前10个主题： [6 25 3 22 26 16 19 4 18 7]
[0.16465983 0.15589012 0.15210117 0.1234063 0.08512253 0.0831406
0.04052934 0.03234385 0.0246687 0.02238315]
第805个文档的前10个主题： [1 25 19 4 10 15 23 28 26 18]
[0.33525038 0.23190863 0.09825045 0.06684136 0.05441141 0.0435245
0.03313123 0.01985919 0.01859455 0.01445226]

主题

每个主题的词分布:

主题#0:

词: 村民 乘客 云南 旅客 地上 裤子 妈妈

概率: [0.00682393 0.0042878 0.00323379 0.00318589 0.00316816 0.00306
0.0029545]

主题#1:

词: 广东省 王某 刘某 辜丸 参议院 榆阳区 陈满

概率: [0.00751067 0.00640128 0.00602311 0.00560491 0.00496156 0.00429384
0.00428849]

主题#2:

词: 工匠 台当局 失误 退役 假如 暴力事件 其一

概率: [0.00298584 0.00257124 0.00240675 0.002152 0.0019788 0.00188708
0.00166307]

主题#3:

词: 李某 充值 工资 毫米 小杰 平均工资 徐某

概率: [0.01106934 0.00335774 0.00318256 0.00301278 0.00299619 0.00285581
0.00280268]

主题#4:

词: 阅读 读书 李 女子 视频 书籍 电子

概率: [0.00996042 0.00583026 0.00567708 0.00562837 0.00477045 0.00399124
0.0039597]

主题#5:

词: 普京 伦敦 俄 会谈 安倍 身份证 被捕

概率: [0.00617236 0.00446138 0.00441558 0.0041976 0.00326962 0.00310108
0.00297686]

主题#6:

词: 企业 政府 患者 公司 医院 建设 医疗

概率: [0.00433506 0.00424583 0.0039865 0.00391137 0.00326307 0.0031044
0.00304006]

路透社数据

159 0:1 2:1 6:1 9:1 12:5 13:2 20:1 21:4 24:2 29:1 35:1 38:2 39:7 48:1 49:1 54:1 59:2 60:1 61:7 66:1
107 0:7 2:2 7:1 16:1 17:1 20:1 24:1 38:3 42:1 59:1 62:1 65:2 70:1 76:2 84:1 87:1 90:2 101:1 107:1 1
153 3:1 4:10 6:4 7:1 8:1 11:9 13:1 20:1 31:3 32:1 33:1 35:2 44:5 45:3 48:5 49:1 62:1 64:1 68:1 71:2
156 0:6 2:1 6:1 7:1 8:1 12:7 18:3 19:1 21:3 22:1 24:3 26:3 27:1 37:1 39:2 40:1 45:1 57:2 60:2 61:3
192 3:2 4:14 5:1 6:1 8:2 9:1 11:11 13:2 14:1 15:3 20:1 26:1 30:1 31:5 33:1 34:1 35:2 37:1 41:1 43:1
180 2:2 3:2 4:24 6:2 8:2 9:1 11:16 13:2 15:2 26:1 31:3 33:3 34:1 35:2 37:3 44:1 48:4 49:1 57:3 64:1
147 3:2 4:7 5:1 6:1 8:1 11:5 13:1 14:1 15:1 31:1 32:1 33:2 34:1 35:2 37:1 41:1 44:4 45:1 48:2 49:2
0 UK: Prince Charles spearheads British royal revolution. LONDON 1996-08-20
184 2:2 3:2 4:20 6:2 8:3 9:1 11:15 13:1 15:1 21
1 GERMANY: Historic Dresden church rising from WW2 ashes. DRESDEN, Germany 1996-08-21
163 1:1 3:2 4:17 5:2 6:2 11:14 13:2 14:2 26:1 2
2 INDIA: Mother Teresa's condition said still unstable. CALCUTTA 1996-08-23
187 0:2 2:2 5:2 7:1 9:3 12:11 14:1 16:1 18:1 13
3 UK: Palace warns British weekly over Charles pictures. LONDON 1996-08-25
170 0:2 3:1 4:1 7:1 12:15 15:2 18:3 19:1 20:1 4
4 INDIA: Mother Teresa, slightly stronger, blesses nuns. CALCUTTA 1996-08-25
224 0:1 2:4 4:3 5:1 6:2 7:2 8:2 9:1 10:3 13:1 5
5 INDIA: Mother Teresa's condition unchanged, thousands pray. CALCUTTA 1996-08-25
193 0:1 1:1 2:1 3:1 4:10 6:2 7:3 8:2 11:10 13:7
6 INDIA: Mother Teresa shows signs of strength, blesses nuns. CALCUTTA 1996-08-26
180 0:1 1:1 2:1 3:1 4:12 6:1 7:2 8:2 11:12 13:8
7 INDIA: Mother Teresa improves, nuns pray for "miracle". CALCUTTA 1996-08-26
237 0:1 2:4 6:2 7:1 8:1 9:2 10:1 12:11 15:1 18
9 UK: Charles under fire over prospect of Queen Camilla. LONDON 1996-08-26
195 0:1 2:1 4:1 12:5 15:1 18:5 19:1 21:3 22:2 10
10 UK: Britain tells Charles to forget Camilla. LONDON 1996-08-27
194 0:2 2:3 3:1 5:1 6:1 7:3 9:1 12:17 15:4 18:11
11 COTE D'IVOIRE: FEATURE - Quiet homecoming for reprieved Ivory Coast maid. ABIDJAN 1996-08-28
165 0:1 3:1 5:1 7:3 12:5 15:3 19:1 20:1 21:2 213
12 INDIA: Mother Teresa (I want to go home") sits and prays. CALCUTTA 1996-08-28
134 0:4 5:1 6:2 9:1 15:1 18:1 19:1 23:3 26:1 314
14 UK: Prosaic end for marriage of Charles and Diana. LONDON 1996-08-28
193 0:3 1:1 2:1 3:1 6:3 8:1 9:2 10:1 13:2 14:215
15 UK: No respite for British royals despite divorce. LONDON 1996-08-28
177 0:4 2:2 5:1 6:3 8:1 9:3 13:1 14:1 15:2 16:16
16 UK: Camilla, love of Charles' life, an unlikely queen. LONDON 1996-08-28
180 2:2 3:1 8:1 14:1 17:6 19:1 34:1 36:3 41:117
17 UK: Diana sets out on new life as single woman. LONDON 1996-08-28
113 0:1 3:1 5:1 6:1 9:1 15:1 30:1 36:1 37:2 4219
18 USA: U.S. Cardinal Bernardin has one year or less to live. CHICAGO 1996-08-30
93 0:3 4:1 5:1 7:4 9:1 14:1 15:1 19:1 20:1 24:20
20 USA: U.S. Cardinal Bernardin says has terminal cancer. CHICAGO 1996-08-30
166 0:2 1:11 3:2 5:2 6:2 7:2 10:1 14:5 15:1 1822
21 ROMANIA: German architect wins Bucharest rebuilding prize. BUCHAREST 1996-09-02
22 ARGENTINA: Argentina's "Blond Angel" finally quits Navy. BUENOS AIRES, Argentina 1996-09-02
23 UK: Disney lights up Pocahontas resting place. GRAVESEND, England 1996-09-06
24 HUNGARY: POPE LEAVES HUNGARY AFTER DEMANDING TWO-DAY VISIT. BUDAPEST 1996-09-07
25 HUNGARY: Pope says mass in Hungary, health in spotlight. GYOR, Hungary 1996-09-07
26 UK: Prince Charles' love will not wed him, paper says. LONDON 1996-09-09

church
pope
years
people
mother
last
told
first
world
year
president
teresa
charles
catholic
during
life
u. s
city
public
time
since
family
king
former
british
harriman
against
country
vatican
made
three
hospital



LDA

```
C:\Python27\python.exe D:/Python/16.3.reuters.py
```

```
type(X): <type 'numpy.ndarray'>
```

```
shape: (395, 4258)
```

```
[[ 1  0  1  0  0  0  1  0  0  1]
 [ 7  0  2  0  0  0  0  1  0  0]
 [ 0  0  0  1 10  0  4  1  1  0]
 [ 6  0  1  0  0  0  1  1  1  0]
 [ 0  0  0  2 14  1  1  0  2  1]
 [ 0  0  2  2 24  0  2  0  2  1]
 [ 0  0  0  2  7  1  1  0  1  0]
 [ 0  0  2  2 20  0  2  0  3  1]
 [ 0  1  0  2 17  2  2  0  0  0]
 [ 2  0  2  0  0  2  0  1  0  3]]
```

```
type(vocab): <type 'tuple'>
```

```
len(vocab): 4258
```

```
('church', 'pope', 'years', 'people', 'mother', 'last', 'told', 'first', 'world', 'year')
```

```
type(titles): <type 'tuple'>
```

```
len(titles): 395
```

```
('0 UK: Prince Charles spearheads British royal revolution. LONDON 1996-08-20', '1 GERMANY:
```

```
LDA start ----
```

```
INFO:lda:n_documents: 395
```

```
INFO:lda:vocab_size: 4258
```

```
INFO:lda:n_words: 84010
```

```
INFO:lda:n_topics: 20
```

```
INFO:lda:n_iter: 500
```

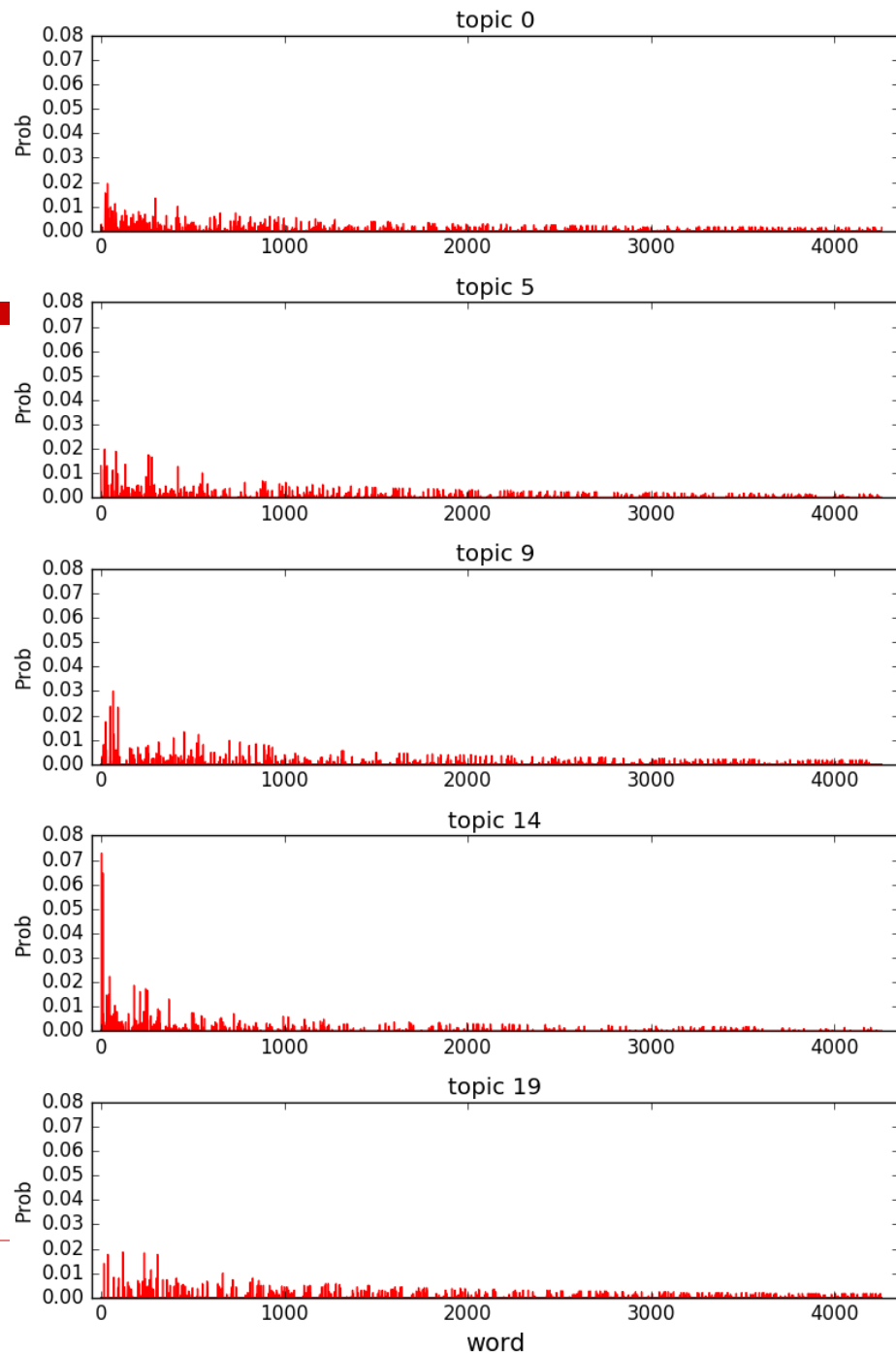
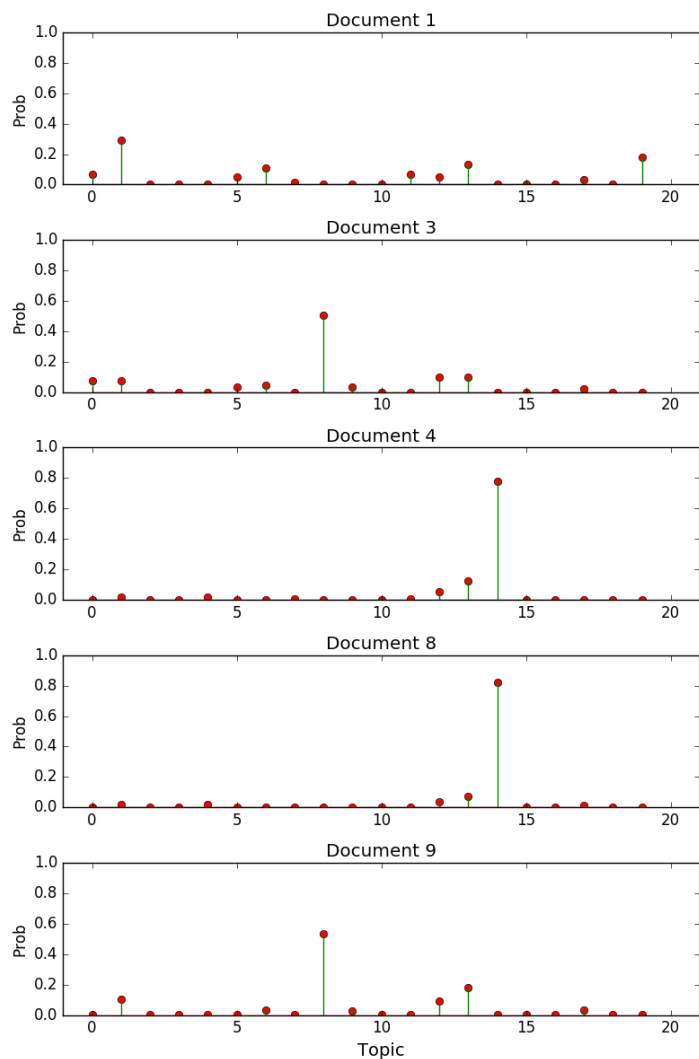
```
INFO:lda:<0> log likelihood: -1051748
```

```
INFO:lda:<10> log likelihood: -719800
```

```
INFO:lda:<20> log likelihood: -699115
```

```
INFO:lda:<30> log likelihood: -689370
```

主题和主题分布

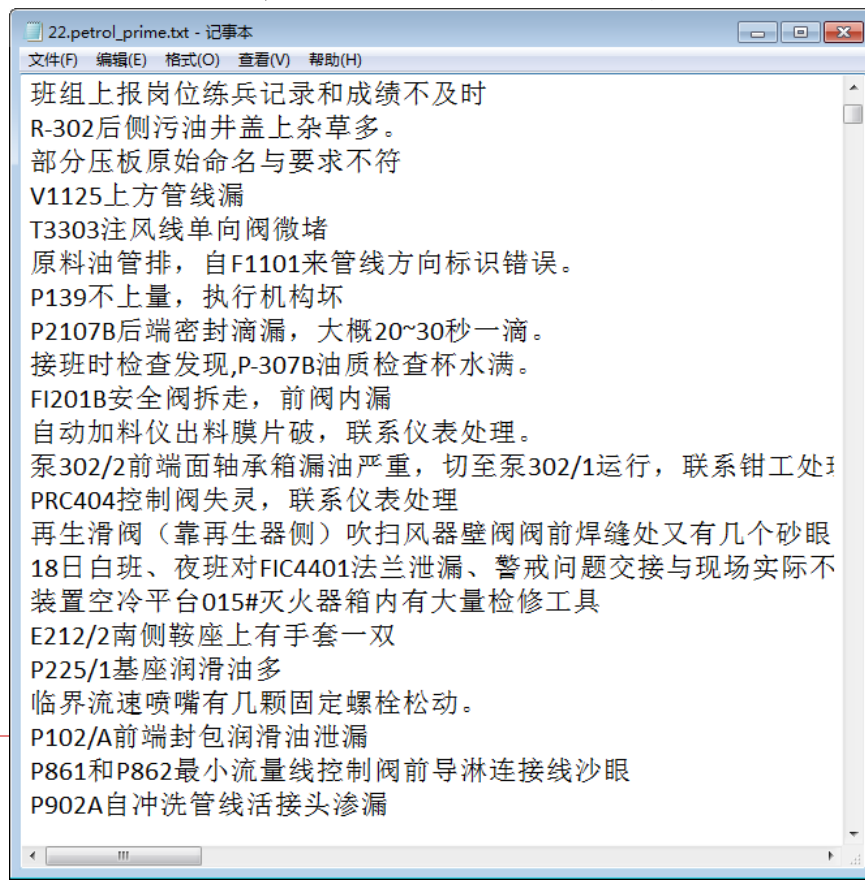


石油例检结果处理

□ 针对国内某石油企业的例行检查处理结果，
试通过主题模型方案，分析例检结果中最突出的问题是什么？

■ 文本共4700个，

■ 单个文档十数字



聚类 “主主题”

22.4.petrol 22.4.petrol

每个主题的词分布:

主题#0: 溶脱 地面 下 发现 溜 吹 漏洞

概率: [0.03567895 0.0305515 0.02995125 0.02905559 0.02847266 0.02672661 0.026

主题#1: 卫生 清扫 新增 本月 缺陷 无 空调

概率: [0.09127588 0.04710893 0.03864091 0.0385492 0.03507678 0.03243568 0.021

主题#2: 过滤器 设备 高 差压 少 入口 17

概率: [0.07382152 0.04735673 0.03770307 0.03342033 0.03160451 0.02609018 0.023

主题#3: 号牌 位 脱落 铁皮 北侧 塔 规范

概率: [0.06217475 0.06146815 0.04165677 0.03554779 0.03128229 0.02976478 0.024

主题#4: 松 建议 液位 损坏 皮带 区域 断裂

概率: [0.03845036 0.03260667 0.03243315 0.031336 0.03074858 0.03062344 0.028

主题#5: 盘根 漏 阀 出口 采样 引出 内

概率: [0.09527604 0.08630304 0.07457675 0.0508139 0.04410229 0.03406847 0.033

主题#6: 日 月 8 红线 压力表 技术员 23

概率: [0.04320296 0.04268257 0.04081915 0.03381051 0.02013684 0.01981086 0.016

主题#7: 地沟 错误 单向阀 螺丝 装车 旁 年

概率: [0.03169998 0.02547323 0.02350422 0.02196816 0.02146054 0.02122409 0.019

主题#8: 皮带 断 不准 松动 一次 盖 被

概率: [0.15231247 0.10833394 0.05339595 0.04884857 0.04585281 0.03503216 0.034

主题#9: 电机 杂音 接头 大盖 冲洗 有 活

概率: [0.07620952 0.0712979 0.06082735 0.03423004 0.03310958 0.02874415 0.027

主题#10: 蒸汽 砂眼 管线 法兰 漏 前有

概率: [0.04160303 0.03960238 0.03123862 0.0293527 0.02685108 0.02634925 0.024

主题#11: 泄漏 伴热 量 牌 入口 底 法兰

概率: [0.07545736 0.05467109 0.05165556 0.03689506 0.03672955 0.03457064 0.031

主题#12: 密封 平台 保温 脱开 缺失 堵头 汽提

概率: [0.04631792 0.04157294 0.0396999 0.03857663 0.03751675 0.03526295 0.034

主题#13: 后端 长明灯 无法 号 灭火器 器 油站

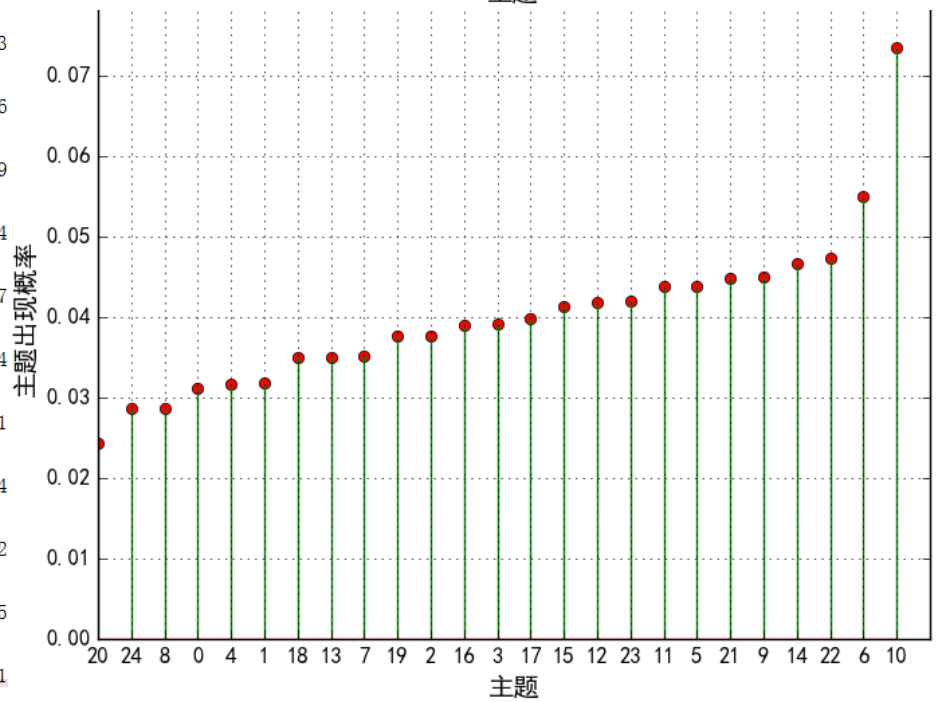
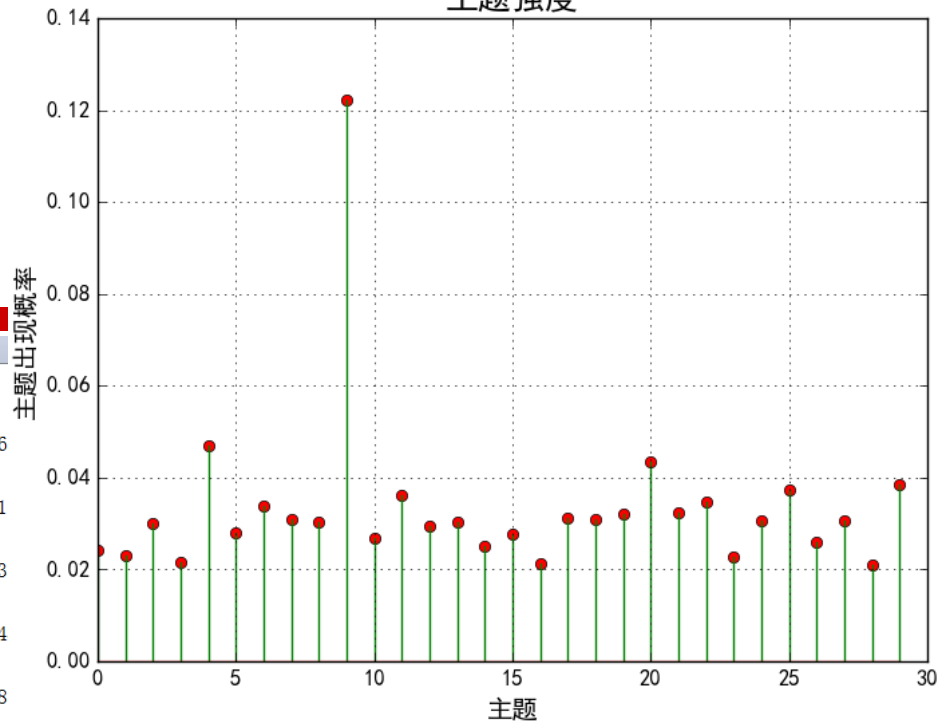
概率: [0.04065264 0.03001566 0.02572 0.02520191 0.02510101 0.02370398 0.022

主题#14: 坏 炉 压力表 润滑油 泵 安全阀 清理

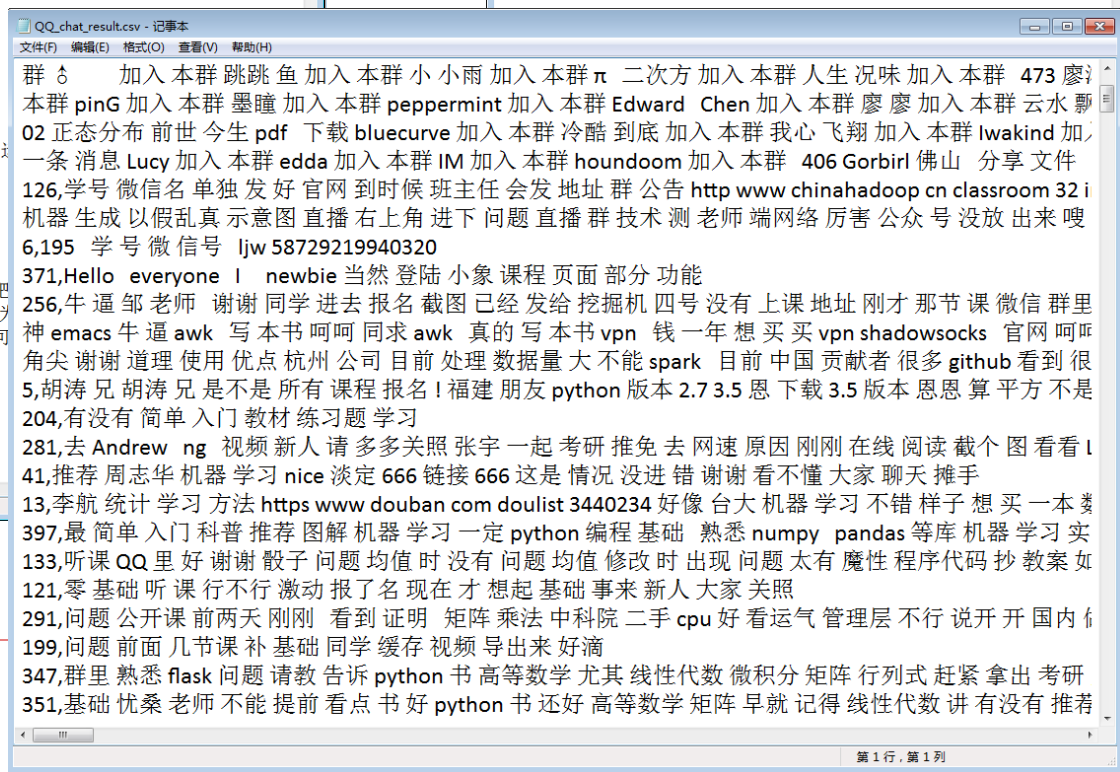
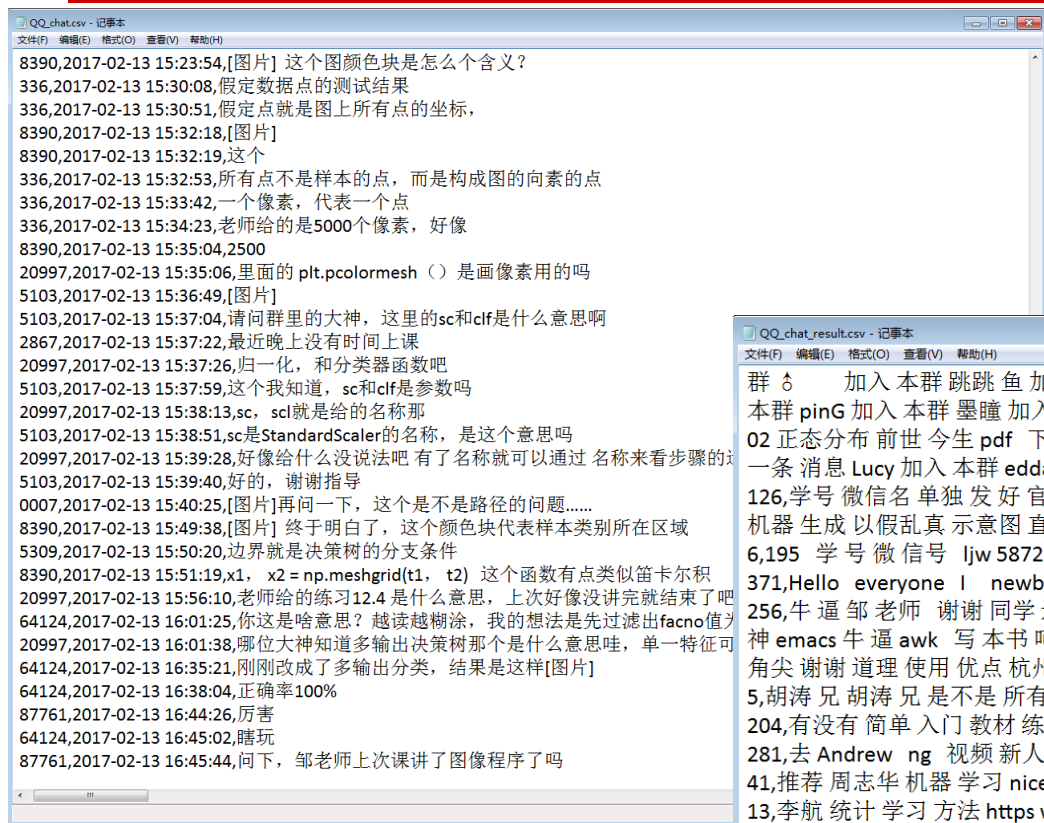
概率: [0.04789169 0.03913219 0.03672254 0.03227133 0.03054908 0.02618981 0.025

主题#15: 缺陷 新增 无 本月 交接班 胶带机 日志

概率: [0.07364203 0.07360244 0.06751279 0.06417833 0.03489421 0.02254108 0.021



聊天记录分析感兴趣话题



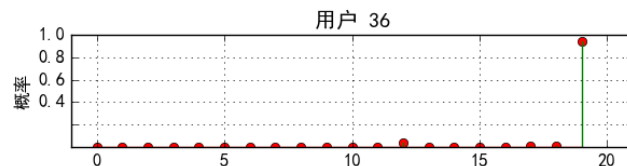
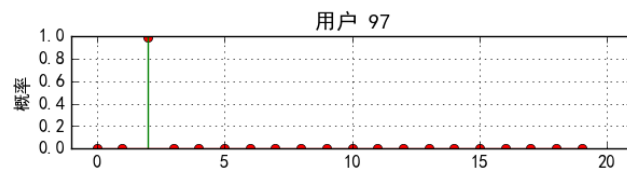
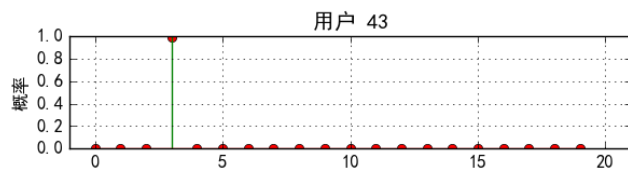
数据处理流程

- 获取QQ群聊天记录: txt文本格式(图1)
 - 整理成“QQ号/时间/留言”的规则形式
 - 正则表达式
 - 清洗特定词: 表情、@XX
 - 使用停止词库
 - 获得CSV表格数据(图2)
 - 合并相同QQ号的留言
 - 长文档利于计算每人感兴趣话题(图3)
 - LDA模型计算主题
 - 调参与可视化
 - 计算每个QQ号及众人感兴趣话题

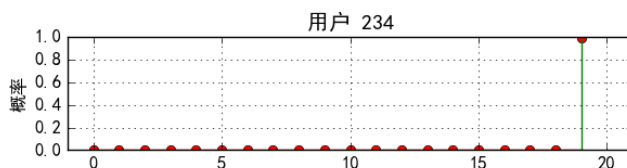
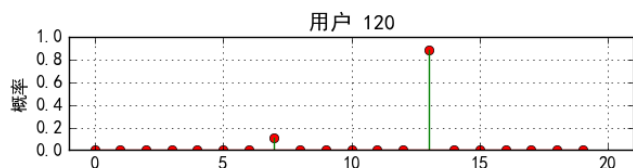
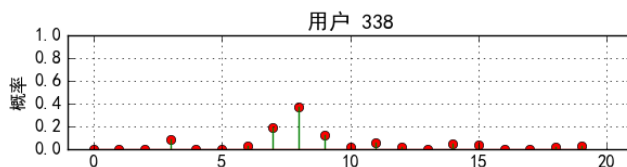
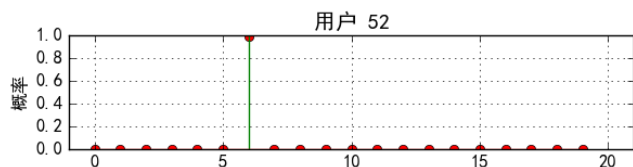
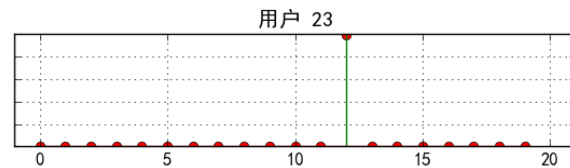
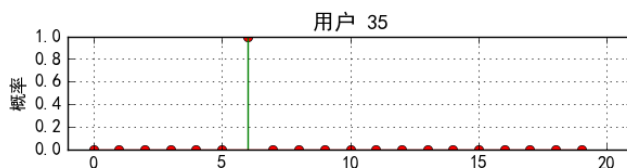
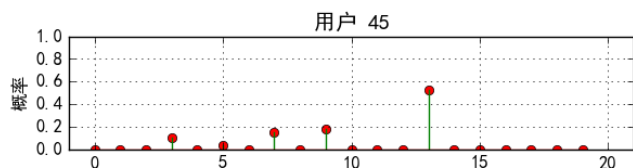
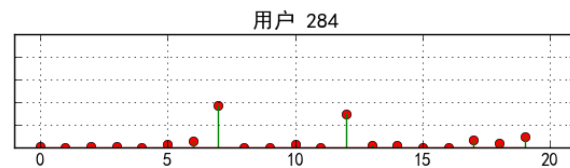
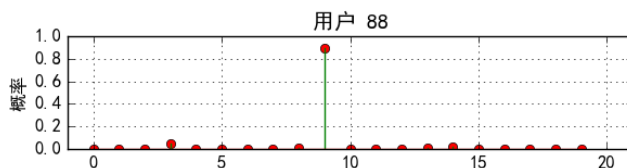
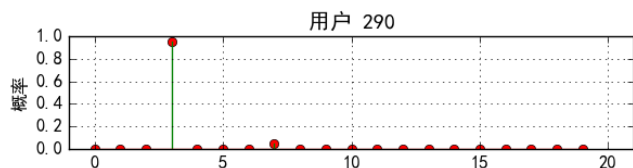
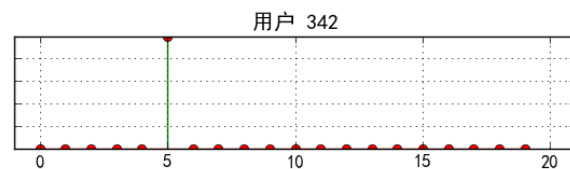
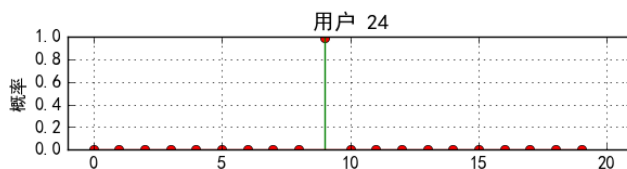
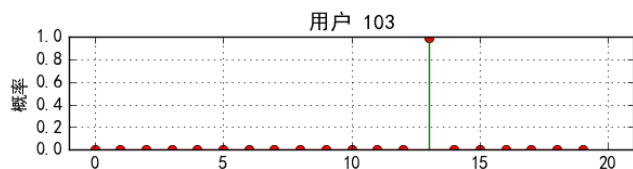


主题分布

用户的主题分布



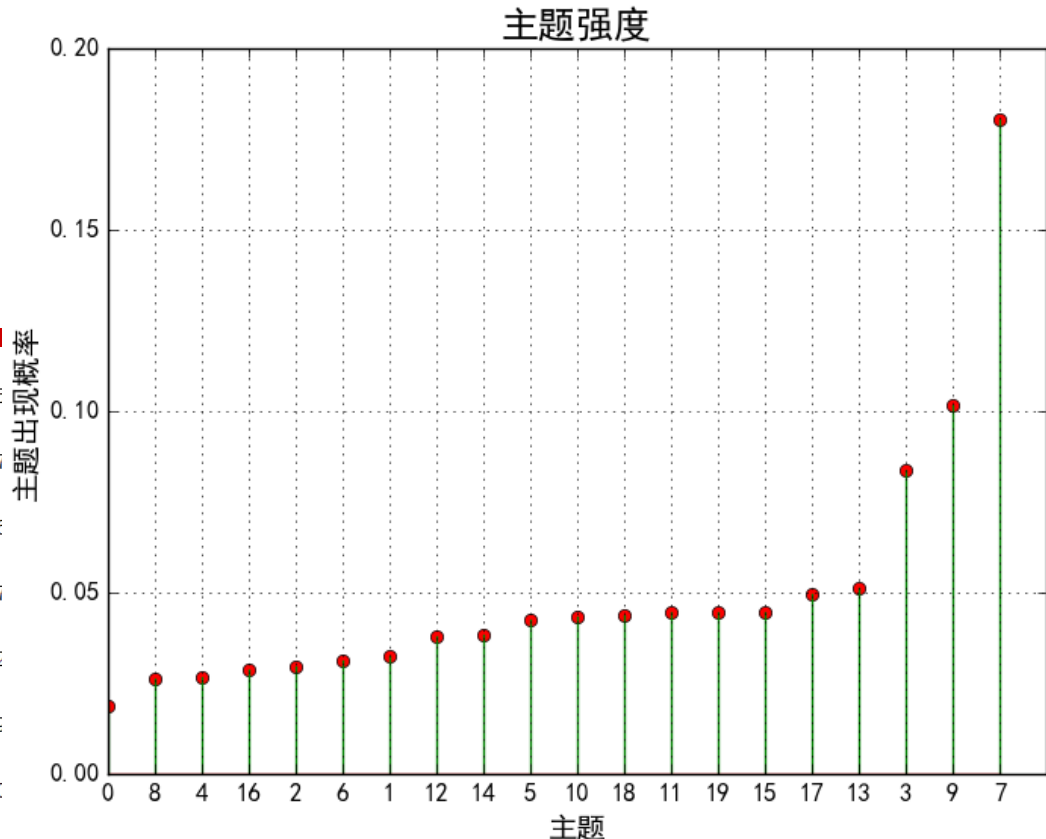
用户的主题分布



主题

感兴趣话题

主题#1:	下周 用到 绑定 matlab 手册 详细描述 666
概率:	[0.00386318 0.00379716 0.00275161 0.00267257 0.0025779 0.0025779 0.0025779 0.0025779 0.0025779 0.0025779]
主题#2:	决策树 下载 代码 新入 记得 无法 if
概率:	[0.00459593 0.00368178 0.00332368 0.00322059 0.00282207 0.00282207 0.00282207 0.00282207 0.00282207 0.00282207]
主题#3:	下载 视频 app 中 cn 电脑
概率:	[0.01650318 0.00643555 0.00534376 0.00493769 0.00477131 0.00477131 0.00477131 0.00477131 0.00477131 0.00477131]
主题#4:	互扫 上装 停留 一份 层次 缺 找点
概率:	[0.00457096 0.00322019 0.00314435 0.00283468 0.0027911 0.0027911 0.0027911 0.0027911 0.0027911 0.0027911]
主题#5:	同问 变量 极限 训练 毕竟 问
概率:	[0.00799138 0.00796053 0.00353821 0.00350646 0.00323556 0.00323556 0.00323556 0.00323556 0.00323556 0.00323556]
主题#6:	上网 元素 等待 白屏 中国 在线看
概率:	[0.00678803 0.00387909 0.00340132 0.00335809 0.00303858 0.00303858 0.00303858 0.00303858 0.00303858 0.00303858]
主题#7:	好 大家 老师 学习 没有 谢谢
概率:	[0.02435813 0.01137977 0.01116381 0.01048067 0.01000842 0.01000842 0.01000842 0.01000842 0.01000842 0.01000842]
主题#8:	参考书 需不需要 离散 型 基础知识 组成
概率:	[0.00909812 0.00337894 0.00242403 0.00229328 0.00220396 0.00215262 0.00215262 0.00215262 0.00215262 0.00215262]
主题#9:	com 9lpintuan wx9c061d2aed9ae09 group https 请
概率:	[0.01010167 0.00642439 0.00517232 0.00517232 0.00495921 0.00485321 0.00467808 0.00467808 0.00467808 0.00467808]
主题#10:	画面 端 找回 QQ 群 账号
概率:	[0.00641685 0.00384681 0.00359311 0.00355134 0.00343251 0.00288641 0.00272948 0.00272948 0.00272948 0.00272948]
主题#11:	直播 今天 一片空白 没有 进不去 问题
概率:	[0.01035459 0.00352562 0.00291557 0.00276692 0.00275323 0.00259631 0.00257422 0.00257422 0.00257422 0.00257422]
主题#12:	太过分 编程 讲完 闪照 配置 这部分
概率:	[0.00510845 0.00290508 0.00289446 0.00273205 0.00265899 0.00251825 0.00249154 0.00249154 0.00249154 0.00249154]
主题#13:	pydotplus 学号 积分 浏览器 弄 请问
概率:	[0.01348479 0.00478899 0.00342937 0.00260794 0.00236339 0.00226667 0.00213521 0.00213521 0.00213521 0.00213521]
主题#14:	分享 两个 满足 话 京东 需求
概率:	[0.0143761 0.00353596 0.00270139 0.00264514 0.00255197 0.00249422 0.00244258 0.00244258 0.00244258 0.00244258]
主题#15:	相等 听不见 openCV 版 数据库 设置
概率:	[0.01212347 0.00449718 0.0030489 0.00300018 0.00285451 0.00281434 0.00263035 0.00263035 0.00263035 0.00263035]
主题#16:	回放 课程 OpenStack 牛云 投放 点挂
概率:	[0.01010629 0.00425028 0.0026812 0.00238546 0.00221883 0.00221883 0.00221883 0.00221883 0.00221883 0.00221883]
主题#17:	福 敬业 早 断 厉害 歌



正则表达式

语法	说明	表达式实例	完整匹配的字符串
字符			
一般字符	匹配自身	abc	abc
.	匹配任意除换行符"\n"外的字符。 在DOTALL模式中也能匹配换行符。	a.c	abc
\	转义字符, 使后一个字符改变原来的意思。 如果字符串中有字符*需要匹配, 可以使用\"或者字符集[*]。	a\.c a\\c	a.c a\\c
[...]	字符集 (字符类)。对应的位置可以是字符集中任意字符。 字符集中的字符可以逐个列出, 也可以给出范围, 如[abc]或[a-c]。第一个字符如果是^则表示取反, 如[^abc]表示不是abc的其他字符。 所有的特殊字符在字符集中都失去其原有的特殊含义。在字符集中如果要使用]、-或^, 可以在前面加上反斜杠, 或把]、-放在第一个字符, 把^放在非第一个字符。	a[bcd]e	abe ace ade
预定义字符集 (可以写在字符集[...]中)			
\\d	数字: [0-9]	a\\dc	a1c
\\D	非数字: [^\\d]	a\\Dc	abc
\\s	空白字符: [<空格>\\t\\r\\n\\f\\v]	a\\sc	a c
\\S	非空白字符: [^\\s]	a\\Sc	abc
\\w	单词字符: [A-Za-z0-9_]	a\\wc	abc
\\W	非单词字符: [^\\w]	a\\Wc	a c
数量词 (用在字符或(...)之后)			
*	匹配前一个字符0或无限次。	abc*	ab abccc
+	匹配前一个字符1次或无限次。	abc+	abc abccc
?	匹配前一个字符0次或1次。	abc?	ab abc
{m}	匹配前一个字符m次。	ab{2}c	abbc
{m,n}	匹配前一个字符m至n次。 m和n可以省略: 若省略m, 则匹配0至n次; 若省略n, 则匹配m至无限次。	ab{1,2}c	abc abbc
*? +? ?? {m,n}?	使 * + ? {m,n} 变成非贪婪模式。	示例将在下文中介。	

边界匹配 (不消耗待匹配字符串中的字符)			
^	匹配字符串开头。 在多行模式中匹配每一行的开头。	^abc	abc
\$	匹配字符串末尾。 在多行模式中匹配每一行的末尾。	abc\$	abc
\\A	仅匹配字符串开头。	\\Aabc	abc
\\Z	仅匹配字符串末尾。	abc\\Z	abc
\\b	匹配\\w和\\W之间。	a\\b!bc	a!bc
\\B	[^\\b]	a\\Bbc	abc
逻辑、分组			
	代表左右表达式任意匹配一个。 它总是先尝试匹配左边的表达式, 一旦成功匹配则跳过匹配右边的表达式。 如果 没有被包括在()中, 则它的范围是整个正则表达式。	abc def	abc def
(...)	被括起来的表达式将作为分组, 从表达式左边开始每遇到一个分组的左括号'(', 编号+1。 另外, 分组表达式作为一个整体, 可以后接数量词。表达式中的 仅在该组中有效。	(abc){2} a(123 456)c	abcaabc a456c
(?P<name>...)	分组, 除了原有的编号外再指定一个额外的别名。	(?P<id>abc){2}	abcaabc
\\<number>	引用编号为<number>的分组匹配到的字符串。	(\\d)abc\\1	1abc1 5abc5
(?P=name)	引用别名为<name>的分组匹配到的字符串。	(?P<id>\\d)abc(?P=id)	1abc1 5abc5
特殊构造 (不作为分组)			
(?...)	(...)的不分组版本, 用于使用' '或后接数量词。	(?:abc){2}	abcaabc
(?i msux)	i msux的每个字符代表一个匹配模式, 只能用在正则表达式的开头, 可选多个。匹配模式将在下文中介。	(?i)abc	AbC
(?#...)	#后的内容将作为注释被忽略。	abc(?#comment)123	abc123
(?=...)	之后的字符串内容需要匹配表达式才能成功匹配。 不消耗字符串内容。	a(?=\\d)	后面是数字的a
(?!...)	之后的字符串内容需要不匹配表达式才能成功匹配。 不消耗字符串内容。	a(?!\\d)	后面不是数字的a
(?<=...)	之前的字符串内容需要匹配表达式才能成功匹配。 不消耗字符串内容。	(?<=\\d)a	前面是数字的a
(?<!...)	之前的字符串内容需要不匹配表达式才能成功匹配。 不消耗字符串内容。	(?<!\\d)a	前面不是数字的a
(?(id/name)yes-pattern no-pattern)	如果编号为id/别名为name的组匹配到字符, 则需要匹配yes-pattern, 否则需要匹配no-pattern。 no-pattern可以省略。	(\\d)abc?(1 \\d abc)	1abc2 abcaabc

常用正则表达式

- ❑ 匹配中文字符: `[\u4e00-\u9fa5]`
- ❑ 匹配双字节字符(包括汉字在内): `[^\x00-\xff]`
- ❑ 匹配空白行: `\n\s*\r`
- ❑ 匹配HTML标记: `<(\S*?)[^>]*>.*?</\1>|<.*? />`
- ❑ 匹配首尾空白字符: `^\s*|\s*$`
- ❑ 匹配Email地址: `\w+([-+.] \w+)*@\w+([-.] \w+)*\.\w+([-.] \w+)*`
- ❑ 匹配网址URL: `[a-zA-z]+://[^\s]*`
- ❑ 匹配帐号合法(5-16位, 字母开头, 允许字母数字下划线): `^[a-zA-Z][a-zA-Z0-9_]{4,15}$`
- ❑ 匹配国内电话号码: `\d{3}-\d{8}|\d{4}-\d{7}`
- ❑ 匹配腾讯QQ号: `[1-9][0-9]{4,}`
- ❑ 匹配中国邮政编码: `[1-9]\d{5}(?! \d)`
- ❑ 匹配身份证: `\d{15}|\d{18}|\d{17}[xX]`
- ❑ 匹配ip地址: `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

常用正则表达式

□ 匹配特定数字：

- 匹配正整数：`^[1-9]\d*$`
- 匹配负整数：`^-[1-9]\d*$`
- 匹配整数：`^-?[1-9]\d*$`
- 匹配非负整数(正整数 + 0)：`^[1-9]\d*|0$`
- 匹配非正整数(负整数 + 0)：`^-[1-9]\d*|0$`
- 匹配正浮点数：`^[1-9]\d*\.\d*|0\.\d*[1-9]\d*$`
- 匹配负浮点数：`^-([1-9]\d*\.\d*|0\.\d*[1-9]\d*)$`
- 匹配浮点数：`^-?([1-9]\d*\.\d*|0\.\d*[1-9]\d*|0?\.\d*|0)\d*$`
- 匹配非负浮点数(正浮点数 + 0)：`^[1-9]\d*\.\d*|0\.\d*[1-9]\d*|0?\.\d*|0$`
- 匹配非正浮点数(负浮点数 + 0)：`^-([1-9]\d*\.\d*|0\.\d*[1-9]\d*)|0?\.\d*|0$`

□ 匹配特定字符串：

- 匹配由26个英文字母组成的字符串：`^[A-Za-z]+$`
- 匹配由26个英文字母的大写组成的字符串：`^[A-Z]+$`
- 匹配由26个英文字母的小写组成的字符串：`^[a-z]+$`
- 匹配由数字和26个英文字母组成的字符串：`^[A-Za-z0-9]+$`
- 匹配由数字26个英文字母或下划线组成的字符串：`^\w+$`

Code

```
def segment():
    stopwords = load_stopwords()
    data = pd.read_csv('QQ_chat.csv', header=0)
    for i, info in enumerate(data['Info']):
        info_words = []
        words = jieba.cut(info)
        for word in words:
            if word not in stopwords:
                info_words.append(word.encode('utf-8'))
        if info_words:
            data.iloc[i, 2] = ' '.join(info_words)
        else:
            data.iloc[i, 2] = np.nan
    data.dropna(axis=0, how='any', inplace=True)
    data.to_csv('QQ_chat_segment.csv', sep=',', header=True, index=False)

def combine():
    data = pd.read_csv('QQ_chat_segment.csv', header=0)
    data['QQ'] = pd.Categorical(data['QQ']).codes
    f_output = open('QQ_chat_result.csv', mode='w')
    f_output.write('QQ,Info\n')
    for qq in data['QQ'].unique():
        info = ' '.join(data[data['QQ'] == qq]['Info'])
        str = '%s,%s\n' % (qq, info)
        f_output.write(str)
    f_output.close()
```

```
def clean_info(info):
    replace_str = (('\\n', ''), ('\\r', ''), ('\\t', ' '), ('表情', ''))
    for rs in replace_str:
        info = info.replace(rs[0], rs[1])

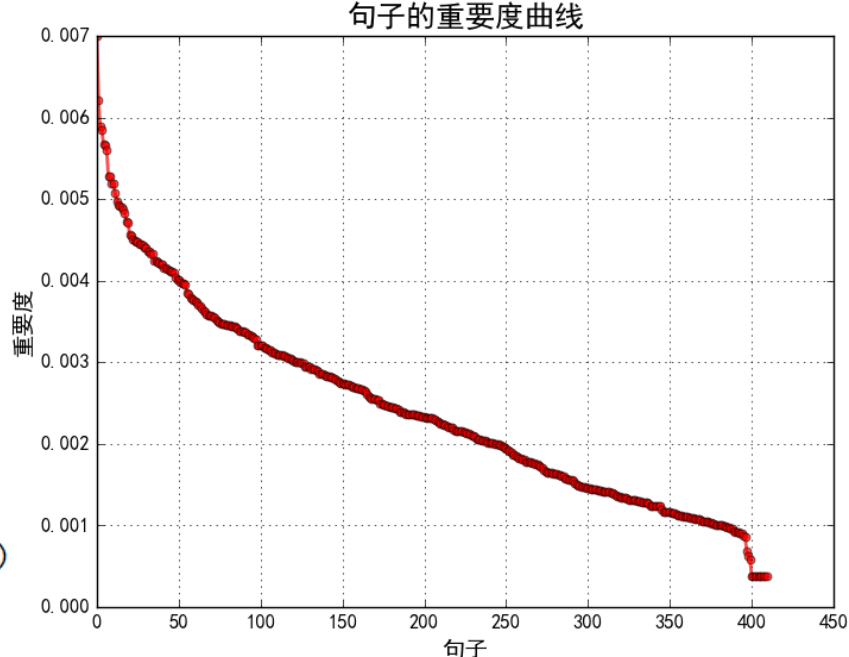
    at_pattern = re.compile(r'(@.* )')
    at = re.findall(pattern=at_pattern, string=info)
    for a in at:
        info = info.replace(a, '')
    idx = info.find('@')
    if idx != -1:
        info = info[:idx]
    return info

def regularize_data():
    time_pattern = re.compile(r'\d{4}-\d{2}-\d{2} \d{1,2}:\d{1,2}:\d{1,2}')
    qq_pattern1 = re.compile(r'([1-9][0-9]{4,})') # QQ号最小是10000
    qq_pattern2 = re.compile(r'(\w+([-+.] \w+)*@\w+([-+.] \w+)*\. \w+([-+.] \w+)*)')
    f = open(u'《机器学习》升级版III.txt')
    f_output = open('QQ_chat.csv', mode='w')
    f_output.write('QQ,Time,Info\n')
    qq = chat_time = info = ''
    for line in f:
        line = line.strip()
        if line:
            t = re.findall(pattern=time_pattern, string=line)
            qq1 = re.findall(pattern=qq_pattern1, string=line)
            qq2 = re.findall(pattern=qq_pattern2, string=line)
            if (len(t) >= 1) and ((len(qq1) >= 1) or (len(qq2) >= 1)):
                if info:
                    info = clean_info(info)
                    if info:
                        info = '%s,%s,%s\n' % (qq, chat_time, info)
                        f_output.write(info)
                        info = ''
                    if len(qq1) >= 1:
                        qq = qq1[0]
                    else:
                        qq = qq2[0][0]
                        chat_time = t[0]
                else:
                    info += line
    f.close()
    f_output.close()
```

Text Rank

```
tr4s = TextRank4Sentence()
tr4s.analyze(text=text, lower=True, source = 'no_stop_words')
data = pd.DataFrame(data=tr4s.key_sentences)
mpl.rcParams['font.sans-serif'] = [u'SimHei']
mpl.rcParams['axes.unicode_minus'] = False
plt.figure(facecolor='w')
plt.plot(data['weight'], 'ro-', lw=2, ms=5, alpha=0.7)
plt.grid(b=True)
plt.xlabel(u'句子', fontsize=14)
plt.ylabel(u'重要度', fontsize=14)
plt.title(u'句子的重要度曲线', fontsize=18)
plt.show()

key_sentences = tr4s.get_key_sentences(num=20, sentence_min_len=4)
for sentence in key_sentences:
    print sentence['weight'], sentence['sentence']
```



0.00699560759634 她知道我心里的苦闷，知道不该阻止我出去走走，知道我要是老呆在家里结果会更糟，但她又担心我一个人在那荒僻的园子里整天都想些什么

0.00621160375013 这一来你中了魔了，整天都在想哪一件事可以写，哪一个人可以让你写成小说

0.00588860912528 那时她的儿子，还太年轻，还来不及为母亲想，他被命运击昏了头，一心以为自己是世上最不幸的一个，不知道儿子的不幸在母亲那儿总是要加倍的

0.00584459738866 她想，只要儿子能活下去哪怕自己去死呢也行，可她又确信一个人不能仅仅是活着，儿子得有一条路走向自己的幸福

0.00567083997126 我奇怪这么小的孩子怎么一个人跑来这园子里

0.00565208315006 我那时脾气坏到极点，经常是发了疯一样地离开家，从那园子里回来又中了魔似的什么话都不说

0.00559372837107 如今我摇着车在这园子里慢慢走，常常有一种感觉，觉得我一个人跑出来已经玩得太久了

0.00527989619912 而且我想，他的母亲也比我的母亲运气好，他的母亲没有一个双腿残废的儿子，否则事情就不这么简单

0.00527906358787 年年月月我都到这园子里来，年年月月我都要想，母亲盼望我找到的那条路到底是什么

0.00519622569726 那天你又说你不如死了好，你的一个朋友劝你：你不能死，你还得写呢，还有好多好作品等着你去写呢

0.00519145626625 他的衣着过分随便，走路的姿态也不慎重，走上五六十米路便选定一处地方，一只脚踏在石凳上或土埂上或树墩上，解下腰间的酒瓶，解酒瓶的当儿

0.00507970004724 她一个人在园子里走，走过我的身旁，走过我经常呆的一些地方，步履茫然又急迫

0.00497335014554 “在那段日子里——那是好几年长的一段日子，我想我一定使母亲作过了最坏的准备了，但她从来没有对我说过：“你为我想想”

0.0049360646412 我便又不能在家里呆了，又整天整天独自跑到地坛去，心里是没头没尾的沉郁和哀怨，走遍整个园子却怎么也想不通：母亲为什么就不能再多活两年

0.00491815078362 是中魔了，我走到哪儿想到哪儿，在人山人海只寻找小说，要是有一种小说试剂就好了，见人就滴两滴看他是不是一篇小说，要是有一种小说显

0.00490464531034 我在这园子里坐着，我听见园神告诉我，每一个有激情的演员都难免是一个人质

0.00486932833768 十五年前的一个下午，我摇着轮椅进入园中，它为一个失魂落魄的人把一切都准备好了

0.00483241065578 有一天我在这园子碰见一个老太太，她说：“哟，你还在这儿哪

0.0047216969869 我才想到，当年我总是独自跑到地坛去，曾经给母亲出了一个怎样的难题

0.00470900507196 我带着本子和笔，到园中找一个最不为人打扰的角落，偷偷地写

word2vec

- Word2vec本质是建立了3层神经网络，将所有词都映射为一定长度(如200)的向量；取一定的窗口范围作为当前词的邻域，估计窗口内的词。

■ 词潜入

- 实验中使用2015年5月爬取的网页新闻作为输入文本，使用gensim的Word2vec包训练词向量。

词典

词典中词的个数： 19542

石块 办公会 基建 最后 着眼于 土 商户 郑重 组织纪律 沿街 岛内 基廷 周靖 工地 圈 律师 极乐 农村土地 期望 广东 央广 人身权利 圆 亲情 集群 眼角 给出 男子 合理 难看 事 城市 伊拉克 亏 于 云 升空 井 区长 为期 些 亚 千克 认定书 青年报 雅安 亡 考察 农业部 未曾 赞同 交 直至 亦 营房 打动 亨 京 亮 检 默契 人畜 亲 中央政治局常委 疲 这时 灵璧县 外人 前往 修房 运动会 中纪委 烂尾 更改 延庆 收购 租房 政商 外事 申请 误工费 建议 租户 主战 宪政 抽屉 张嘉伟 开工 首席 入院 夺下 力争 出卖场 爆 唐 讨 安全事故 明确 姐姐 解决方案 雁过拔毛 前方 战斗机 心思 工作证 国民政府 陈水扁 铁轨 浅 有权 购物中心 覆盖面 地势 停诊 热潮 侵占 心疼 因素 比哈尔邦 响起 身患 开启 才 律师函 行贿 寄出 冷战 区位 宣判 拽 不至于 莱西 分析 形象 遮拦 是因为 剖析 药房 顿时 穿过 过半 录取 两难 秀屿 越发 客户端 巡视 拉票 鄱阳县 PAKDA 独家 玉皇 埃塞俄 丢弃 斐然 境界 反馈 净利润 政权 风格 文明 最起码 常会 情节 事端 经查 谈及 生命 涉外 恨不得 民事裁定 独立王国 违法 美发 战争 配 文工团 票据 A 素 党纪国法 版本 拐 放火 运走 儿媳妇 劳埃德 实验室 相应 行政长官 求医 此番 有 领土问题 负伤 广度 候选人 牵动 最 校长 不够 疾病 望 不备 朝 期 总政治部 配置 累 人行道 朗 打 斗 木 学院 业态 魏先生 这种 监禁 角度 王明 口罩 一致 黄唇 学文 贩毒 高原 企图 炮塔 捍卫 过世 鼓吹 咱们 巡逻机 产 货物 攻坚 APP 车位 郑绍鑫 读卖新闻 啤酒瓶 其后 秦玉海 松井 北极圈 十几年来 张北县 纽约市 北京电视台 美国共和党 砖 要出 动员 boyangcongpeople 砍 每天晚上 大军 亲朋 想见 平方米 荷兰 消防员 名头 笃信 先后 生计 填报 蛋白质 供认不讳 管窥 阵营 坤叔 房屋 储备 川 组织部 收集 屈某 校外 重新启动 策略 十月 屋外 差 外包 姚某 世界各地 左 房山 指导 郝纯毅 伟大 视为 已 风景区 并行 巴 铁道 作 纠葛 起点 汽油 否认 农业局 裁定 愿望 集团军 是从 重工 北京市公安局 工作作风 立法 非正式 点儿 海报 协和医院 咳嗽 打电话 乐队 此举 全局 姑息 日台 男童 拥堵 勘察 俗 发到 法律文书 论述 老屋 战胜 马克思主义 逝者 编译 坦率地 遇上 体外 无期 西湖 带动 冲毁 产物 委派 上当受骗 去世 此后 大厂县 中国佛教协会 保护者 提取 互信 可怕 提 二维码 吴艾莉 施工方 选举 张通荣 哪位 分流 人事变动 想出 出发 冈比亚 面前 二甲基 性行为 厂房 军长 陶瓷 应有 储蓄 申冤 土耳其政府 签字 守候 马伊 成分 学业 边检 预 带队 放在 冷孟梅 爱心 袭击 韩某 定案 不明真相 预定 宋先生 办公厅 术语 会长 鱼鹰 背后 向永林 雨量 防微杜渐 创始人 王庄 以往 康定 news 衢州市 二妹 交通管理 跳车 哥 九州 网讯 焦兰生 其中 油漆 师范大学 问起 买不起 前两天 高洪顺 成都市委 失事 造势 其一 高尔夫球 信息中心 开区 平时 拒收 其三 国门 隔壁 商务通 人大常委会 资产 复杂 亲戚朋友 安全监管 应该 我行我素 非要 其次 总监 藏品 晏涛 庄河市 咖啡 新西兰 毛发 分开 遗留问题 控股 治病 经济社会 官僚 深刻 纵容 汉口 进一步 真经 力气 视频 脱团 湖南省委 购 眼线 深切 木材 留给 处警 大字 初步统计 增长极 自强 大概 砍掉 确为 轰炸 商家 丁学君 原因 近些年 等级 被害人 打仗 心怀 遗物 协调一致 奶 反对者 木板 曹 负责 适用 外来 数以亿计 清江 更何况 桥 落到实处 大方 军委 县市区 字词 案 年纪 机翼 阐明 桌 桂 龙岩 检方 受到限制 层级 女 法定程序 父母 洪金洲 陈 节点 该处 何等 克制 二环路 碎 生病 摄制组 电梯门 时会 碗 测量 知情人 打开 主管 成人 碘 威廉 时说 碧 联合国大会 沙子 慰问 远远 火爆 夺走 激情 性别 切实 击碎 火箭 樊某 碰 区县 收 站不住脚 毛利率 进而 数学 幸福感 环保部门 五月 十日 冰块 遗址 股份制 远东地区 数字 进军 小山头 薛凤 捡起 Facebook 两学 开善 产出 科长 护栏 昨夜 免税 今日 医健 勇 夏伯渝 已经 图案 就医 可谓 原判 王纪平 整理 奈何 下跌 阅读 亲友 脏器 暴涨 三叉戟 考上 包庇 不懈努力 脱口秀 难料 鼓楼 金融时报 空气质量 救灾 积极主动 萨吉姆 有效 礼仪 确信 斗鱼 停在 发愁 手中 弊端 不雅 承接 美国大学 底特律 商业秘密 战线 砖混 道德 艰苦奋斗 月入 取暖 放出 合法权利 癌变 医疗事故 此人 德州 产科 手上 手下 体育 爱情 所知 相对来说 患 光环 副作用 恶劣 悦 之一 工程学院 安抚 您 之上 贡献力量 悬 建国 严厉 E 振源 宣泄 巨大损失 发帖人 天津 华夏 痛恨 放管 民事 小心 复决 三板 治 美国国务院 锅 锁 偷盗 补发 不出 回来 半岛 随身 刑警大队 残留量 钻营 用心 应酬 医疗保险 键 盎司 纪念日 蹲点 不减 站上 更多 饮片 王强 卡号 不准 潜入 刘建军 普通人 集中统一 张贴 假警察 质量 真诚 贷款 事前 山洪 研究生 甲乙双方 刑警 高手 韩军 敲竹勒索罪 仓库 普查 背上 标注 鹿邑县 臭味 随身携带 俄联邦 医用 朱明国 擒获 发觉 黄 外交部 皇冠 形成 桂江 尘埃落定 有害信息 分拆 墨西哥 怀上 脱逃 益康 再三 脑子 领跑 堂而皇之 永久性 不对劲 读者 漂亮 主治医师 MA 酒席 细胞 姑娘 灵 成绩 伤残 灰 灯 热度 白 纯粹 车门 上面 车间 嘴 机长 市场机制 阿里 欧内斯特 元素 公职人员 巴西利亚 国务院办公厅 欠下 难忍 夜晚 缓和 前兆 这里 规划图 酣睡 发育 集约 巨额 海宁 波兰 表格 三死一伤 爱浜 操办 招募 卢某诚 刘会文 警戒线 友善 年迈 算账 药业 约见 名录 恒频 长春 刘峻成 相距 二氧化硫 满脸 通了 卡通 更是 的 的 会面 填 上来 布满 外侧 信 干部队伍 准确 排放 那种 根源 辐射 党性原则 农副产品 有违 将军 幕府 紧锁 行列 桥梁 成功 勇气 大雨 推动 杂物 华裔 推演 通知书 物种 江苏队 电警棍 郊区 核准 通畅 免

训练结果

中国 和 祖国 的相似度为: 0.532070
中国 和 毛泽东 的相似度为: 0.312460
中国 和 人民 的相似度为: 0.371986
祖国 和 毛泽东 的相似度为: 0.882594
祖国 和 人民 的相似度为: 0.759524
毛泽东 和 人民 的相似度为: 0.583692

苹果 三星 美的 海尔 离群词: 海尔
中国 日本 韩国 美国 北京 离群词: 北京
医院 手术 护士 医生 感染 福利 离群词: 福利
爸爸 妈妈 舅舅 爷爷 叔叔 阿姨 老婆 离群词: 舅舅

中国 + 城市 - 学生 :

经济 0.842732787132
国际 0.842595815659
战略 0.839224457741
发展 0.816590726376
我国 0.792158007622
全球 0.782959520817
推动 0.781142055988
军事 0.778445780277
台海 0.776162207127
税制 0.7729408741

男 + 工作 - 女 :

两学 0.74424546957
做好 0.723003566265
督察 0.708856344223
党校 0.68696641922
电视电话会议 0.679121792316
各级 0.677579462528
明确提出 0.668009757996
中央 0.664826393127
充分认识 0.661263227463
谈话 0.659905314445

俄罗斯 + 美国 + 英国 - 日本 :

法国 0.944284677505
安倍 0.94345831871
菲律宾 0.923454284668
俄 0.923405408859
朝鲜 0.921864390373
普京 0.919387102127
敌对 0.918260753155
大选 0.915652871132
国民党 0.911968111992
主流 0.911087334156

与 学习 最相近的词:

结构性 0.948330461979
侧 0.94113522768
创新 0.939934253693
树立 0.939579963684
注重 0.933748006821
发挥 0.93070846796
精益求精 0.925910234451
人才 0.924418389797
供给 0.923088252544
党校 0.92102253437

与 公安局 最相近的词:

分局 0.98934006691
大队 0.975338459015
支队 0.963695168495
鹿邑县 0.961336493492
西城 0.95579791069
刑警 0.955546677113
公安分局 0.953128516674
隆化县 0.952146053314
控申 0.951352596283
专案组 0.950993955135

与 大学 最相近的词:

秦川 0.889399170876
中央党校 0.88586807251
中科院 0.856984019279
学历 0.856915056705
吉安 0.84980738163
研究生 0.847551643848
优秀 0.846781611443

参考文献

- David M. Blei, Andrew Y. Ng, Michael I. Jordan, *Latent Dirichlet Allocation*, 2003
- Gregor Heinrich, *Parameter estimation for text analysis*. 2008
- Matthew D. Hoffman, David M. Blei, Francis Bach. *Online learning for Latent Dirichlet Allocation*. 2010
- http://en.wikipedia.org/wiki/Dirichlet_distribution
- http://en.wikipedia.org/wiki/Conjugate_prior

我们在这里

□ <http://wenda.ChinaHadoop.cn>

■ 视频/课程/社区

□ 微博

■ @ChinaHadoop

■ @邹博_机器学习

□ 微信公众号

■ 小象学院

■ 大数据分析挖掘

互联网新技术在线教育领航者

小象问答 搜索标题、用户 搜索 全站内容搜索 提问 首页 动态 发现 话题 通知

全部 招聘求职 机器学习 大数据平台技术 DCon 大数据行业应用 NoSQL数据库 数据科学 江湖救急

发现 最新 推荐 热门 等待回复

graphviz has no attribute 'write' 贡献
邹博 回复了问题 • 2 人关注 • 1 个回复 • 3 次浏览 • 2017-04-09 15:48

sklearn中如何理解Pipeline机制 贡献
数据分析与数据挖掘 邹博 回复了问题 • 2 人关注 • 1 个回复 • 28 次浏览 • 2017-04-09 15:39

关于9.Logistic回归的ppt中第9页的对数线性函数 贡献
机器学习 邹博 回复了问题 • 3 人关注 • 3 个回复 • 39 次浏览 • 2017-04-09 15:35

关于“贝叶斯估计中，最大后验概率估计就是结构化风险最小化的例子：当模型是条件概率分布，损失函数为对数损失函数，模型的复杂度由模型的先验概率表示，结构化风险最小化就等价于最大后验概率估计” 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 26 次浏览 • 2017-04-09 15:27

关于连续值的预测 贡献
咨询 邹博 回复了问题 • 2 人关注 • 1 个回复 • 31 次浏览 • 2017-04-09 15:24

拉格朗日对偶函数为什么一定是凸函数 贡献
数据科学 邹博 回复了问题 • 2 人关注 • 2 个回复 • 26 次浏览 • 2017-04-09 15:20

梯度下降公式中的斯堪J 是 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 29 次浏览 • 2017-04-09 15:17

深度学习适合做预测吗？ 贡献
深度学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 27 次浏览 • 2017-04-09 15:15

关于6.4PCA_FeatureSelection.py中plt.legend的参数疑问 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 28 次浏览 • 2017-04-09 15:04

@邹博 有哪些可以下载数据源的网站？ 贡献
数据分析与数据挖掘 邹博 回复了问题 • 4 人关注 • 1 个回复 • 31 次浏览 • 2017-04-09 14:53

LDA主题模型 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 29 次浏览 • 2017-04-09 14:45

代码10.6bagging_ridged老师提到了采样率设为0.2能够使峰值部分的数据被体现出来。这是为什么呢？ 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 22 次浏览 • 2017-04-09 14:26

GraphViz's executables not found 贡献
机器学习 邹博 回复了问题 • 3 人关注 • 2 个回复 • 23 次浏览 • 2017-04-09 13:47

决策树中关于feature_importances代码的问题 贡献
机器学习 邹博 回复了问题 • 2 人关注 • 1 个回复 • 6 次浏览 • 2017-04-09 13:11

专题
招聘求职
大数据行业应用
数据科学
系统与编程
云计算技术

热门话题 更多 >
机器学习 907 个问题, 230 人关注
spark 387 个问题, 172 人关注
hadoop 1059 个问题, 155 人关注
python数据分析 171 个问题, 28 人关注
数据分析与数据挖掘 54 个问题, 111 人关注

热门用户 更多 >
小心巴 14 个问题, 0 次赞同
又又V 45 个问题, 22 次赞同
铁甲无声 10 个问题, 0 次赞同
带刀锦衣卫 13 个问题, 0 次赞同

感谢大家！

恳请大家批评指正！