

“智慧政务”中的文本挖掘应用

摘要

近年来，随着微信、微博、市长信箱、阳光热线等网络问政平台逐步成为政府了解民意、汇聚民智、凝聚民气的重要渠道。因此，基于自然语言处理技术的智慧政务系统已经是社会治理创新发展的新趋势，对提升政府的管理水平和施政效率具有极大的推动作用。

针对问题一，我们首先对附件 2 中的数据去重，用饼图表示各分类的占比情况，发现各类别数据比值相差不大。将得到的无重复数据中的留言详情列按一级分类提取到 txt 中，利用 jieba 中文分词工具将留言详情的内容进行分词，用 Bunch 将分词后的文本信息转换为文本向量信息并且对象化，并将停用词去除。构建词向量空间，统计文本词频，生成文本的词向量空间，用 TF-IDF 方法权重测量，抽取反映各类别特征词。可以使用朴素贝叶斯分类算法来进行文本分类，按照要求，我们用 F-Score 对分类方法进行评估。

针对问题二，对附件 3，我们统计发言人的频数，选取最高的 11 项，再看发言人发表留言的热度，然后检查其留言内容会不会太过冗长，结合点赞数，因为点赞数能反映出人们较为关注的事件倾向。研究方法 为准确衡量各指标权重，本次研究采用层次分析法对指标进行赋权。其基本思想是把人们 处理复杂体系的定性分析，转化为定性与定量结合的体系分析，用群体判断克服单一判断的主观偏好，进行群体综合，再以定量的形式给出准确的排序结果。

针对问题三，对于答复意见的评价，我们首先从所在地区所占据

的记录数进行分析，再从对应的答复时间间隔进行分析，从而侧面分析人工办事效率的高低，反过来分析哪个地区出现的问题比较多。

关键词：去重 中文分词 jieba 词向量 TF-IDF 方法

F-Score 算法 特征向量法

Text mining application in "smart government"

Abstract

In recent years, with the development of wechat, microblog, mayor's mailbox, sunshine hotline and other online political platforms, the government has gradually become aware of public opinion

An important channel to gather people's wisdom and spirit. Therefore, the smart government system based on natural language processing technology has become a new trend of social governance innovation and development, which plays a great role in promoting the management level and efficiency of the government.

To solve the first problem, we first de duplicate the data in Annex 2, and use pie chart to show the proportion of each category, and find that the ratio of each category is similar. The message details in the non repetitive data are extracted into TXT according to the first level classification. The content of the message details is segmented by using the Chinese word segmentation tool of Jieba. The segmented text information is converted into text vector information and objectified by bunch, and the stop words are removed. Construct the word vector space, count the word frequency of the text, generate the word vector space of the text, use TF-IDF method to measure the weight, extract and reflect the characteristic words of each category. We can use naive Bayesian classification algorithm to classify text. According to the

requirements, we use F-score to evaluate the classification method.

In response to question two, for Annex 3, we count the frequency of speakers, select the top 10 items, and then look at the heat of the speakers' comments, and check whether the content of their comments will be too lengthy, combined with the likes, because the likes can reflect the tendency of events that people are more concerned about. The research method is to accurately measure the weight of each index. In this study, AHP is used to weight the index. Its basic idea is to transform people's qualitative analysis of complex system into a system analysis combining qualitative analysis and quantitative analysis, to overcome the subjective preference of single judgment with group judgment, to carry out group synthesis, and then to give accurate ranking results in quantitative form.

In response to question three, for the evaluation of the response opinions, we first analyze the number of records occupied by the region, and then analyze the corresponding response time interval, so as to analyze the efficiency of manual work, and in turn analyze which region has more problems.

key word: Duplicate removal Chinese participle Jieba

Word vector TF-IDF method F-score algorithm

目录

1. 挖掘目的	7
2. 分析方法与过程	8
2.1 总体流程	8
2.2 详细步骤如下:	8
2.3 问题一分析与过程	9
2.3.1 流程图	9
2.3.2 数据预处理	9
2.3.3 TF-IDF 算法	10
2.3.4 贝叶斯算法	11
2.3.5 使用 F-Score 对分类方法进行评价	11
2.2 问题二分析与过程	12
2.2.1 定义热点问题	12
2.2.2 热点问题数据筛选	12
2.2.3 统计与表格制作分析	14
2.3 问题三分析与过程	16
3.结果分析	17
3.1 问题一实现代码与结果截图	17
3.1.1 下面这段代码将利用 jieba 对语料库进行分类... ..	17
3.2 问题二代码实现与结果截图	22
4.结论	28

5.参考文献	29
--------------	----

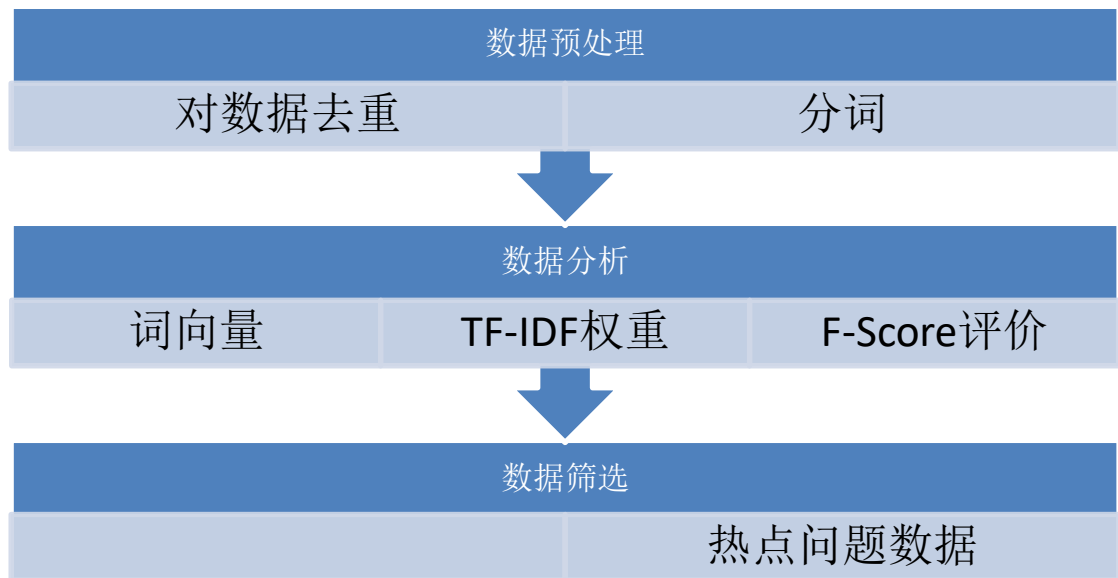
1. 挖掘目的

本次建模目的是通过利用已有的群众问政留言记录，利用自然语言处理和文本挖掘的方法对留言进行分类，涉及到 **jieba** 中文分词工具，**Scikit-Learn** 库的 **Bunch** 数据结构，**TF-IDF** 方法，**F-Score** 算法以达到以下目的：

- a) 对留言详情语料库进行分词对此类非数据结构的数据进行文本挖掘，从训练集生成 **TF-IDF** 向量词袋，利用权重策略得到具有很好的类别区分能力的词，根据结果，可以结合其他数据分析群众问政的热点问题。
- b) 根据模型，对群众留言的测试数据进行分类，预测模型的可行性，缓解只依靠人工的低效率压力。
- c) 针对相关部门对留言的答复意见，从答复的相关性，完整性，可解释性等方面，给出可靠的评价方案。

2. 分析方法与过程

2.1 总体流程



2.2 详细步骤如下:

- 1) 数据预处理，对题目给出的数据做，有很多重复的留言信息，要在原始的数据上对这部分信息进行去重处理；查看每一类别的数据占比过程，防止有些类别数据过少，对分类模型造成较大的偏差，并按需求将数据分类或转换文本格式；在这些基础上将留言详情文本进行中文分词。
- 2) 数据分析，对分词后的信息构建词向量，把不同类别的留言信息用 TF-IDF 方法权重测量，得到每一留言类别的关键词；用朴素贝叶斯分类算法训练分类词器，判断信息的所属类别；：用 F-Score 对分类方法进行评价。
- 3) 数据筛选，由相关信息筛选汇总，找出附件 3 中的热门问题。

- 4) 针对附件 4 相关部门对留言的答复意见，从答复的相关性、完整性、可解释性等角度对答复意见的质量给出一套评价方案，并尝试实现。

2.3 问题一分析与过程

2.3.1 流程图



2.3.2 数据预处理

- 1) 为得到更为严谨的数据，必须要对数据去重操作，去除重复信息。考虑到同一留言用户可能对不同问题提出留言，或是同一时间段有可能有多用户留言等因素，我们重点根据留言主题和留言详情信息去重。

- 2) 中文分词及词向量构建，在数据分析挖掘之前，首先要将数据转换成计算机能够看到的数据，对于非结构类型的文本数据我们使用分词的方法，对大量的留言数据进行分词，为后面提取出能够让计算机根据重点词频进行分类的词汇做准备。

2.3.3 TF-IDF 算法

TF-IDF(Term Frequency-Inverse Document Frequency, 词频-逆文件频率).

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。一个词预测主题的能力越强，权重越大，反之，权重越小。

TF 计算公式

$$TF_{\omega} = \frac{\text{在某一类中词条}\omega\text{出现的次数}}{\text{该类中所有的词条数目}}$$

逆向文件频率 (inverse document frequency, IDF) IDF 的主要思想是：如果包含词条 t 的文档越少，IDF 越大，则说明词条具有很好的类别区分能力。某一特定词语的 IDF，可以由总文件数目除以包含该词语之文件的数目，再将得到的商取对数得到。

IDF 计算公式

$$IDF = \log \left(\frac{\text{语料库的文档总数}}{\text{包含词条}\omega\text{的文档数}+1} \right)$$

某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的 TF-IDF。因此，TF-IDF 倾向于过滤掉常见的词语，保留重要的词语。

TF-IDF 算法计算公式

$$\text{TF-IDF} = \text{TF} * \text{IDF}$$

TF-IDF 值越大，则这个词成为一个关键词的概率就越大。

在对留言详情信息分词后，需要把词汇转换成向量词，采用 TF-IDF 算法可以将留言详情信息转换为权重向量。

2.3.4 贝叶斯算法

贝叶斯分类算法时一大类分类算法的总称。贝叶斯分类算法以样本可能属于某类的概率来作为分类依据。

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

这里我们选择 Scikit-Learn 的朴素贝叶斯算法进行文本分类，测试集随机抽取自训练集中的文档集合，每个分类下的文档。训练步骤：首先是分词，之后生成文件词向量文件，直到生成词向量模型。在训练词向量模型时，需要加载训练集词袋，将测试产生的词向量映射到训练集词袋的词典中，生成向量空间模型。使用多项式贝叶斯算法来进行测试文本分类，返回分类精度。

2.3.5 使用 F-Score 对分类方法进行评价

f- Score 是一种统计量，F-Score 又称为 F-Measure，F-Score 是 Precision 和 Recall 加权调和平均，是 IR（信息检索）领域的常用的一个评价标准，常用于评价分类模型的好坏。

$$F_1 = \frac{1}{n} \sum_{i=1}^n \frac{2P_iR_i}{P_i + R_i}$$

P 是准确率(Precision), R 是召回率(Recall)。

1.准确率 = 系统检索到的相关文件/系统所有检索到的文件总数

2.召回率,即检索出相关文档数和文档库中所有的相关文档数的比率,衡量的是检索系统的查全率。

召回率=系统检索到的相关文件/系统所有相关的文件总数

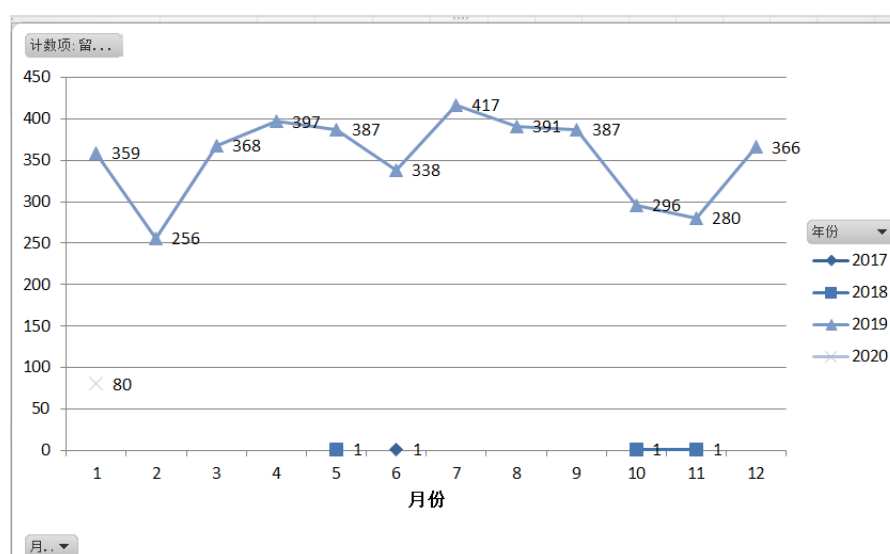
2.2 问题二分析与过程

2.2.1 定义热点问题

根据附件 3 我们有留言用户, 留言主题, 留言时间, 留言详情, 反对数, 点赞数等数据, 经过分析比较, 我们主要就留言用户, 留言时间, 留言详情, 点赞数几方面入手。通过用户热度、内容热度及传播热度定义最终问题热度。

2.2.2 热点问题数据筛选

基于附件 3, 我们用 Excel 表就留言的时间先做一个分析。



此图为附件三根据时间做出来的一个图, 横坐标为各个月份, 纵坐标为各个月份的留言数。我们可以发现数据多基于 2019 年, 根据此图获得 2019 年 1 月份的数据, 对此数据进行分析。因为数据的选取是随机的, 所以只分析 1 月份的, 其余的也可类似分析。

根据文件特点及网络留言的一些特征, 通过用户热度、内容热度及传播热度

三个一级指标衡量热度评价指标，每个二级指标下设计相应的三级指标，如下图所示：

热度评价指标	一级指标	二级指标
	用户热度	发表留言的频度
		留言所属标签热度
	内容热度	内容充实度
	传播热度	点赞数（反对、赞同）

（1）用户热度是与用户在一段时间内在网络上发表留言的次数有关，在判断留言所属一级标签后，根据标签热度再进行筛选

（2）内容热度是从留言内容角度，探究内容的充实度，即内容是否冗长，留言是否言简

（3）传播热度是从留言被点赞的次数出发，选取获赞高的
研究方法

为准确衡量各指标权重，本次研究采用层次分析法对指标进行赋权。层次分析法（AHP）是由美国著名运筹学家萨蒂于上世纪 70 年代创立的，用于解决多目标评价体系中指标权重确定的问题。其基本思想是把人们处理复杂体系的定性分析，转化为定性与定量结合的体系分析，用群体判断克服单一判断的主观偏好，进行群体综合，再以定量的形式给出准确的排序结果

（1）首先通过此代码获取多次留言的用户，并生成一个 dup.csv 文件

```
import pandas as pd
path = 'D:/泰迪杯/C 题/2019-1.xlsx'
xlsx = pd.ExcelFile(path)
df = pd.read_excel(xlsx, 'Sheet1')
df['is_duplicated'] = df.duplicated(['留言用户'])
a = df['is_duplicated'].sum()
print(a)
df_dup = df.loc[df['is_duplicated'] == True]
df_dup.to_csv('dup.csv', encoding='utf-8')
```

```
E:\python3.7\python.exe D:/taidibei/demo04.py
```

```
60
```

```
Process finished with exit code 0
```

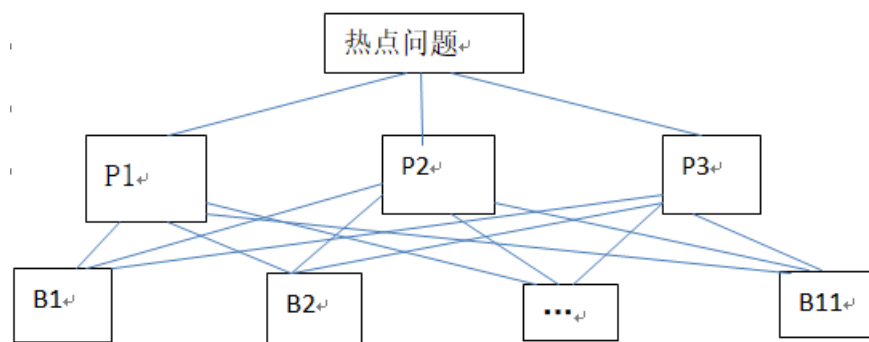
重复名共有 60 条，再根据生成的重复留言用户进行 excel 排查，查看重复用户在这一个月中留言的次数，从中选出频数大于等于 3 的数据，再获取数据所属地点，最后再判断数据所属一级标签，结果如下图：

留言用户	频数	总获赞数	地点/人群	所属一级标签
A00031618	14	20	A 市建设，西地省实验小学	环境保护、城乡建设、教育文体、经济管理、民政
A00085667	4	28	A7 县公路、公交	城乡建设
A000100792	6	3	A 市、A1 区饭店、物业	卫生计生、环境保护
A00035110	3	13	西地省拖欠工资	劳动和社会保障
A00084854	3	8	A 市建设	劳动和社会保障、环境保护、教育文体
A00078096	3	0	A 市供水问题	民政
A00015946	3	2	A5 区电器市场、A3 区市政广场、A 市公交	交通运输、政法
A00080329	3	94	A 市地区规划	经济管理
A00051608	3	0	A 市拆迁	经济管理、城乡建设
A000108466	4	0	A 市公交、旅游	教育文体、商贸旅游
A00013021	3	20	A7 县违规建设	纪检监察

2.2.3 统计与表格制作分析

目标为选择热点问题，问题为上图所涉及的一级标签，现用字母来表示标签：环境保护（B1），城乡建设（B2），教育文体（B3），经济管理（B5），民政（B6），卫生计生（B7），劳动和社会保障（B8），交通运输（B9），政法（B10），商贸旅游（B11），根据频数（p1）、获赞数（p2）、地点/人群（p3）这 3 个准则去反复比较这些标签。

现用层次分析法来解决这个文体，步骤如下：首先将热点问题分为三个层次，最上层为目标层，即热点问题，最下层为方案层，有标签 B1、B2、…、B11，中间层为准则层，有 p1、p2、p3，构建层次图如下：其次确定各准则对于目标的权重及各方案对于每一准则的权重，最后组合各层权重，得到方案层因素对于目标的权重，下面只对准则层各因素对于目标的权重建立一般的数学模型。



设准则层中因素 P_i 对于目标的权重为 ω ， ω 反映了因素 P_i ($i=1, 2, 3$) 相对于目标的重要程度，及 $\omega = (\omega_1, \omega_2, \omega_3)^T$ 其中

$\omega_i > 0$, 且 $\sum_{i=1}^n \omega_i = 1$, $n=5$ ，则 ω 就是各因素的权重向量，这是一个未知量

首先是采用两两比较的方法(或其它有效的方法)构造判断矩阵 A. 具体做法如下：就选择标签这一目标，因素 P_i 与因素 P_j 比较，分别得到 3 个相对分值 ($j=1, 2, 3$)。若 $a_{ij} > 1$ ，则表示 P_i 比 P_j 重要，且重要程度是 P_j 的 a_{ij} 倍； $a_{ij} \leq 1$ ，则表示 P_j 比 P_i 重要，且重要程度是 P_i 的 $1/a_{ij}$ ($j=1, 2, 3$) 倍

令 $A = (a_{ij})_{3 \times 3}$ (其中, $a_{ii} = 1, a_{ij} = \frac{1}{a_{ji}}$, $i, j = 1, 2, 3, 4, 5$), A 称为判

断矩阵, 由这个矩阵的元素可以确定因素 P_i 对于目标的综合分值, 这个值记为 y_i ($i = 1, 2, 3$)。

利用 Excel 表, 结合用户热度、内容热度及传播热度得到的数据, 对热点问题进行降序热度排名, 并表明问题的时间范围, 地点/人群, 问题描述 (具体结果见热点问题表.xls)

2.3 问题三分析与过程

附件四中总共有 2816 个记录, 其中有关 A 市、K 市、B 市、L 市、C 市、J 市以及 G 市的记录分别占 1018 个、293 个、266 个、257 个、153 个、145 个、130 个。其余各市总计记录 554 个。可见 A 市的留言记录占总数 36%, 说明 A 市存在的问题较其他各市多。侧面说明我们应该重点对 A 市的各个方面进行调查以及整改。

而在答复时间间隔方面, 在有关残疾人福利和低保也就是国家扶贫方面, 答复时间间隔长达 1160 天, 说明其处理程度偏难以及处理效率低。其次为生态安葬方面, 两者均属第三标签, 可见第三标签方面的留言较杂, 同时答复时间偏长。

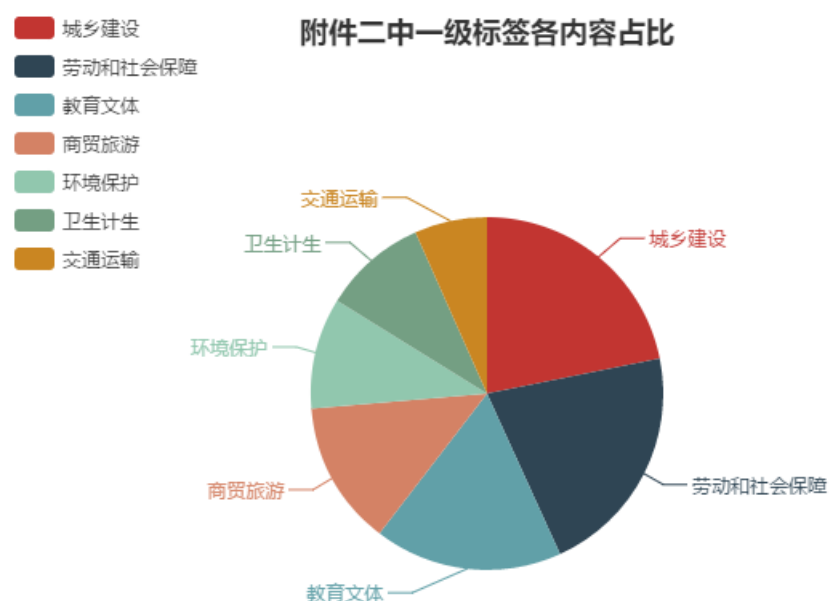
现在我们利用 Excel 电子表格中的 MID() 函数对附件四中的地址以及答复时间间隔的数据进行提取, 其中提取 1030 条记录并利用 Tableau 可视化进行数据分析。

可视化结果图见第三点结果分析的 3.3 部分

3.结果分析

3.1 问题一实现代码与结果截图

用饼图表示各分类的占比情况，排除各类别数据过少而产生较大的误差。

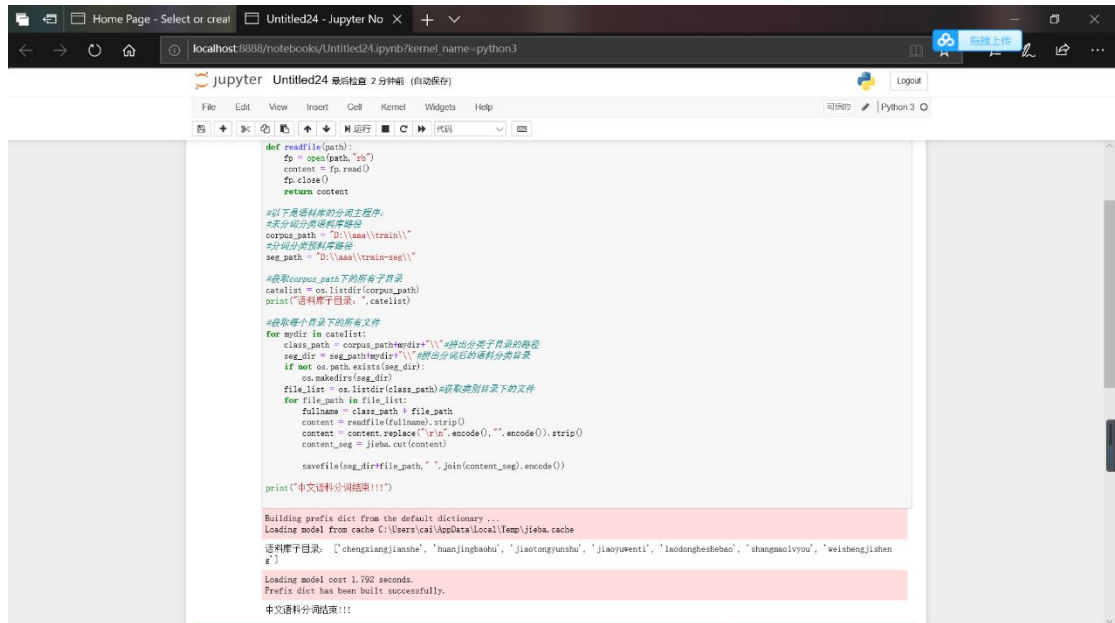


3.1.1 下面这段代码将利用 jieba 对语料库进行分类

语料库截图：

名称	修改日期	类型	大小
chengxiangjianshe	2020/5/5 8:37	文件夹	
huanjingbaohu	2020/5/5 8:40	文件夹	
jiaotongyunshu	2020/5/5 8:42	文件夹	
jiaoyuwent	2020/5/5 8:44	文件夹	
laodongheshebao	2020/5/5 8:45	文件夹	
shangmaolvyou	2020/5/5 8:46	文件夹	
weishengjisheng	2020/5/5 8:47	文件夹	

代码实现分词



```
def readfile(path):
    fp = open(path, "rb")
    content = fp.read()
    fp.close()
    return content

#以下是要处理的文本主程序:
#求分词语料库路径
corpus_path = "D:\\aaa\\train\\"
seg_path = "D:\\aaa\\train-seg\\"

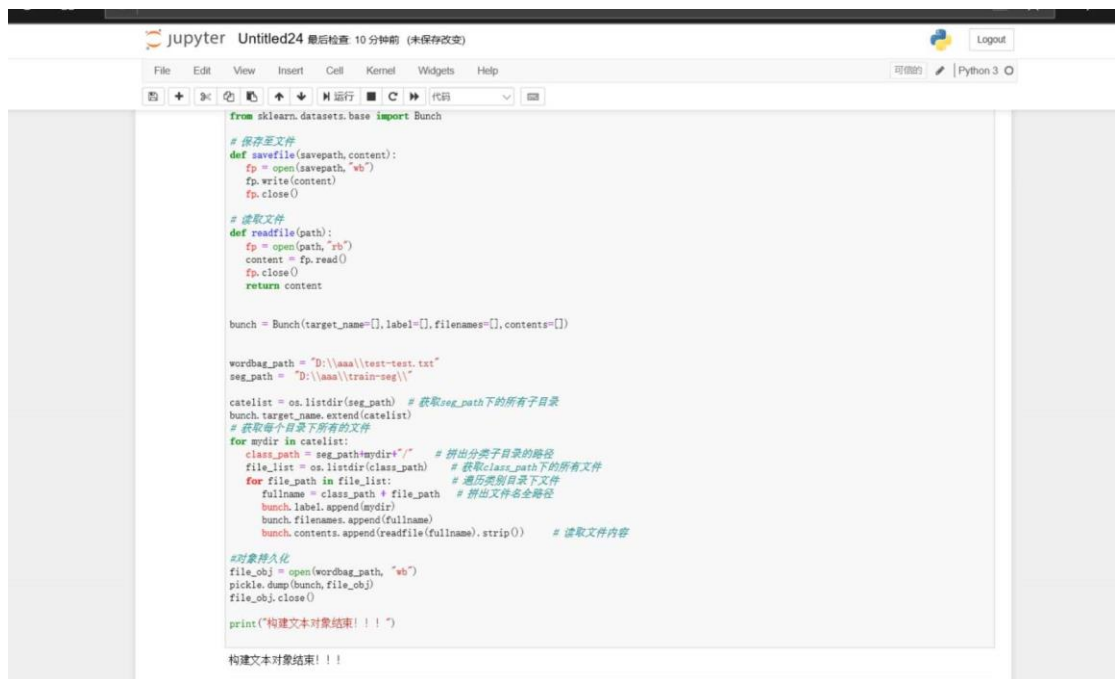
#获取corpus_path下的所有子目录
catalist = os.listdir(corpus_path)
print("语料库子目录:", catalist)

#获取每个目录下的所有文件
for mydir in catalist:
    class_path = corpus_path + mydir + "\\" #拼出分类子目录的路径
    seg_dir = seg_path + mydir + "\\" #拼出分词后的语料分类目录
    if not os.path.exists(seg_dir):
        os.makedirs(seg_dir)
    file_list = os.listdir(class_path) #获取类别目录下的文件
    for file_path in file_list:
        fullname = class_path + file_path
        content = readfile(fullname).strip()
        content = content.replace("\n", "").encode("utf-8").strip()
        content_seg = jieba.cut(content)
        savefile(seg_dir + file_path, ". ".join(content_seg).encode("utf-8"))

print("中文语料分词结束!!!")

Building prefix dict from the default dictionary ...
Loading model from cache C:\Users\cal\AppData\Local\Temp\jieba.cache
语料库子目录: ['chengxiangjianshe', 'huanjingbaohu', 'jiaorongyunshu', 'jiaoyuwent', 'laodongshesheao', 'shangmaolvyou', 'weishengjishen']
Loading model cost 1.792 seconds.
Prefix dict has been built successfully.
中文语料分词结束!!!
```

为了后续生成向量空间模型的方便,这些分词后的文本信息还有转换为文本向量信息并且对象化。这里需要用到 **Scikit-Learn** 库的 **Bunch** 数据结构。



```
from sklearn.datasets.base import Bunch

# 保存至文件
def savefile(savepath, content):
    fp = open(savepath, "wb")
    fp.write(content)
    fp.close()

# 读取文件
def readfile(path):
    fp = open(path, "rb")
    content = fp.read()
    fp.close()
    return content

bunch = Bunch(target_name=[], label=[], filenames=[], contents=[])

wordbag_path = "D:\\aaa\\test-test.txt"
seg_path = "D:\\aaa\\train-seg\\"

catalist = os.listdir(seg_path) # 获取seg_path下的所有子目录
bunch.target_name.extend(catalist)
# 获取每个目录下所有的文件
for mydir in catalist:
    class_path = seg_path + mydir + "\\" # 拼出分类子目录的路径
    file_list = os.listdir(class_path) # 获取class_path下的所有文件
    for file_path in file_list:
        fullname = class_path + file_path # 拼出文件全路径
        bunch.label.append(mydir)
        bunch.filenames.append(fullname)
        bunch.contents.append(readfile(fullname).strip()) # 读取文件内容

#对象持久化
file_obj = open(wordbag_path, "wb")
pickle.dump(bunch, file_obj)
file_obj.close()

print("构建文本对象结束!!!")

构建文本对象结束!!!
```

在文本分类之前会自动过滤掉某些字和词,被过滤掉的词或字称作停用词。这类词一般是意义模糊的常用词,还有一些语气助词,通常对文本起不了分类特征的意义。

读取停用词表的代码


```
File Edit View Insert Cell Kernel Widgets Help 可信的 Python 3
testspace.vocabulary = trainbunch.vocabulary

#创建词频的持久化
space_path = "D:\\aaa\\testspace.txt" #词向量空间保存路径
writebunchobj(space_path, testspace)

In [2]: from sklearn.naive_bayes import MultinomialNB #导入多项式贝叶斯算法包
import pickle

#读取Bunch对象
def readbunchobj(path):
    file_obj = open(path, "rb")
    bunch = pickle.load(file_obj, encoding="utf-8")
    file_obj.close()
    return bunch

#导入训练集向量空间
trainpath = r"D:\\aaa\\wordspace.txt"
train_set = readbunchobj(trainpath)

#导入测试集向量空间
testpath = r"D:\\aaa\\testspace.txt"
test_set = readbunchobj(testpath)

#应用朴素贝叶斯算法
#alpha: 0.001 alpha越小, 迭代次数越多, 精度越高
clf = MultinomialNB(alpha=0.001).fit(train_set.tdm, train_set.label)

#预测分类结果
predicted = clf.predict(test_set.tdm)
total = len(predicted)
rate = 0
for flabel, file_name, expect_cate in zip(test_set.label, test_set filenames, predicted):
    if flabel != expect_cate:
        rate += 1
    print(file_name, ":", 实际类别, ":", flabel, "----> 预测类别:", expect_cate)

print("error rate", float(rate)/float(total), "%")

error rate 0.0 %

In [ ]:
```

分类结果评估

from sklearn.naive_bayes import MultinomialNB #导入多项式贝叶斯算法包

import pickle

from sklearn import metrics

from sklearn.metrics import precision_score #sklearn 中的精准率

#读取 Bunch 对象

def readbunchobj(path):

file_obj = open(path, "rb")

bunch = pickle.load(file_obj, encoding="utf-8")

file_obj.close()

return bunch

定义分类精度函数

def metrics_result(actual, predict):

```

        print("          准          确
率:",metrics.precision_score(actual,predict,average='macro'))

        print("    召    回    率    :",    metrics.recall_score(actual,
predict,average='macro'))

        print("f1-score:",    metrics.f1_score(actual,
predict,average='macro'))

        #导入训练集向量空间

        trainpath = r"D:\\aaa\\wordspace.txt"

        train_set = readbunchobj(trainpath)

        #导入测试集向量空间

        testpath = r"D:\\aaa\\testspace.txt"

        test_set = readbunchobj(testpath)

        #应用朴素贝叶斯算法

        #alpha:0.001 alpha 越小，迭代次数越多，精度越高

        clf = MultinomialNB(alpha= 0.001).fit(train_set.tdm,train_set.label)

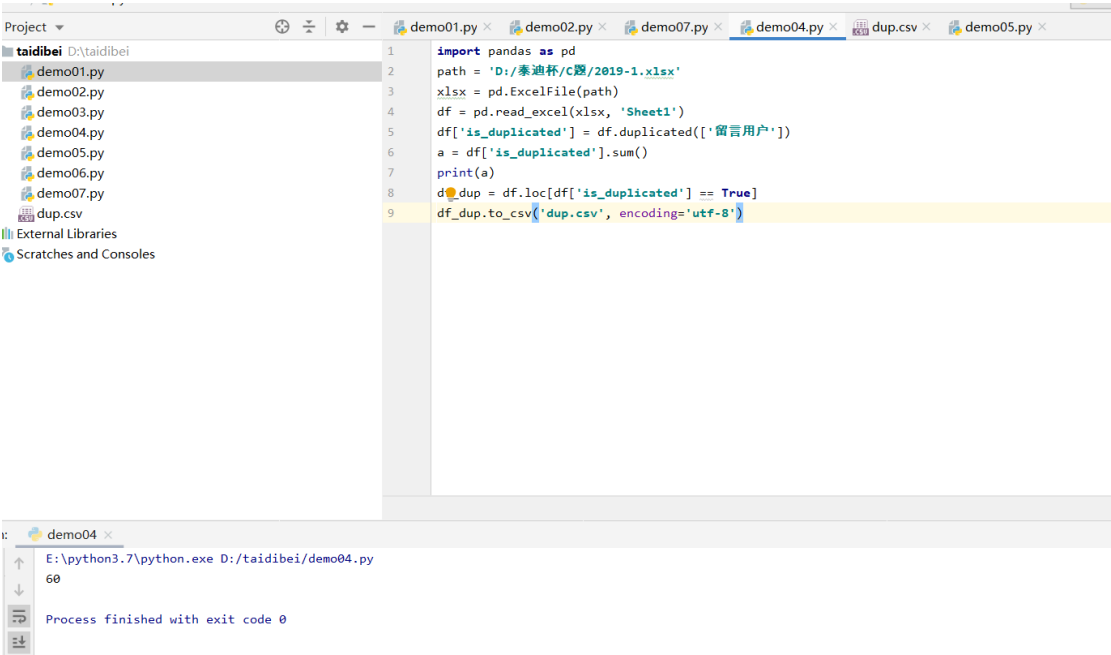
        #预测分类结果

        predicted = clf.predict(test_set.tdm)

        metrics_result(test_set.label,predicted)

```

3.2 问题二代码实现与结果截图



热点问题留言明细表

留言编号	留言用户	留言主题	留言时间	留言详情	点赞数	回复数
197662	A00085667	咨询A7县东四路远大路到人民路南延拆迁进度等问题	2019/1/22 21:08:46	东四路远大路到人民路南延, 2017年已经提上议程, 现在进	0	18
204713	A00085667	请问A7县新安路泉塘公交首末站规划在哪个位置?	2019/1/22 21:23:48	请问A7县新安路泉塘公交首末站规划在哪个位置?	0	4
239905	A00085667	咨询去年4月份公示新增的几条A7县星沙公交线路进度	2019/1/22 21:14:10	去年4月份公示新增的几条星沙公交线路进度, 星沙泉塘	0	5
265931	A00085667	咨询去年A7县新增的两条公交线路进度	2019/1/22 21:15:57	去年四月份公示了新增两条线路, 星沙泉塘(山河智能)-	0	1
233542	A00080329	问问A市经开区东六线以西泉塘和商业中心以南的有	2019/1/2 20:27:26	A市经开区东六线以西, 泉塘和商业中心以南, 新蕾品	0	24
239670	A00080329	问问A市经开区东六线以西泉塘和商业中心以南的有	2019/1/11 15:46:04	A市经开区东六线以西, 泉塘和商业中心以南, 新蕾品	0	41
256358	A00080329	问问A市经开区东六线以西泉塘和商业中心以南的有	2019/1/2 20:27:07	A市经开区东六线以西, 泉塘和商业中心以南, 新蕾品	0	29
212691	A00013021	A7县星沙违建之风盛行无人管	2019/1/17 13:37:47	尊敬的市委书记, 您好, 特向您反映A7县星沙碧桂园威尼	0	2
270731	A00013021	A7县违建之风盛行谁来管?	2019/1/17 13:29:46	A7县碧桂园别墅区大范围的违建事实, 包括但不限于挖地	1	8
278497	A00013021	A7县碧桂园威尼斯城大范围的违建	2019/1/17 13:50:34	A7县碧桂园威尼斯城大范围的违建事实, 包括但不限于挖	0	9
198911	A00035110	西地省山干制药机械股份有限公司拖欠A7县工人工资近	2019/1/7 18:00:45	西地省山干制药机械股份有限公司拖欠工人的工资近半年	0	0
223409	A00035110	西地省山干制药机械股份有限公司拖欠A7县工人工资近	2019/1/7 17:59:37	西地省山干制药机械股份有限公司拖欠工人的工资近半年	0	7
230813	A00035110	西地省山干制药机械股份有限公司拖欠工人工资近半年	2019/1/3 22:05:24	西地省山干制药机械股份有限公司拖欠工人的工资近半年	0	6
190669	A00084854	关于A市财富名园小区相关设施建设的建议	2019/1/8 17:37:48	财富名园小区位于万家丽路和三一大道十字路口的西南角	0	6
224621	A00084854	“A市财富名园空地多年”一帖回复后, 开发商毫无动	2019/1/11 16:25:13	上一年在https://baidu.com/帖子中得到反馈, 可是开?	0	0
255772	A00084854	A市财富名园空地多年, 严重影响市容市貌	2019/1/11 16:09:42	财富名园小区位于万家丽路和三一大道十字路口的西南角	0	2

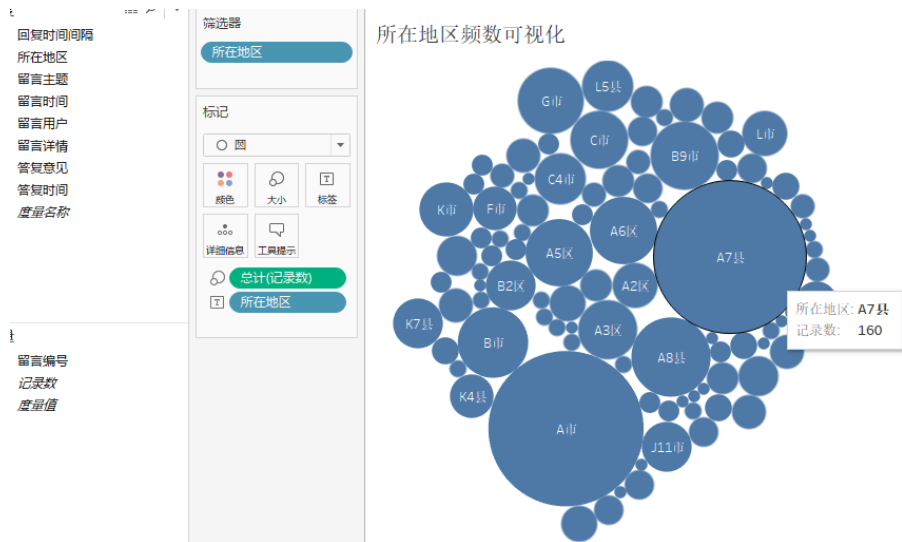
热点问题表

热度排名	问题ID	热度指数	时间范围	地点/人群	问题描述
1	1	7	2019. 1. 1-2019. 1. 31	A7县公路、公交	道路的拆迁、公交建设
2	2	6	2019. 1. 1-2019. 1. 31	A市	询问对闲置场地的规划
3	3	4	2019. 1. 1-2019. 1. 31	A7县违建建设	A7县违建之风盛行
4	4	3	2019. 1. 1-2019. 1. 31	西地省拖欠工资	拖欠工人工资进半年
5	5	3	2019. 1. 1-2019. 1. 31	A市建设	土地闲置, 设施建设差

注: 热度指数10为满

3.3 问题三分析可视化结果截图

首先对所在地区和回复时间间隔数据进行可视化分析



回复时间间隔
所在地区
留言主题
留言时间
留言用户
留言详情
答复意见
答复时间
度量名称

筛选器
所在地区
度量名称

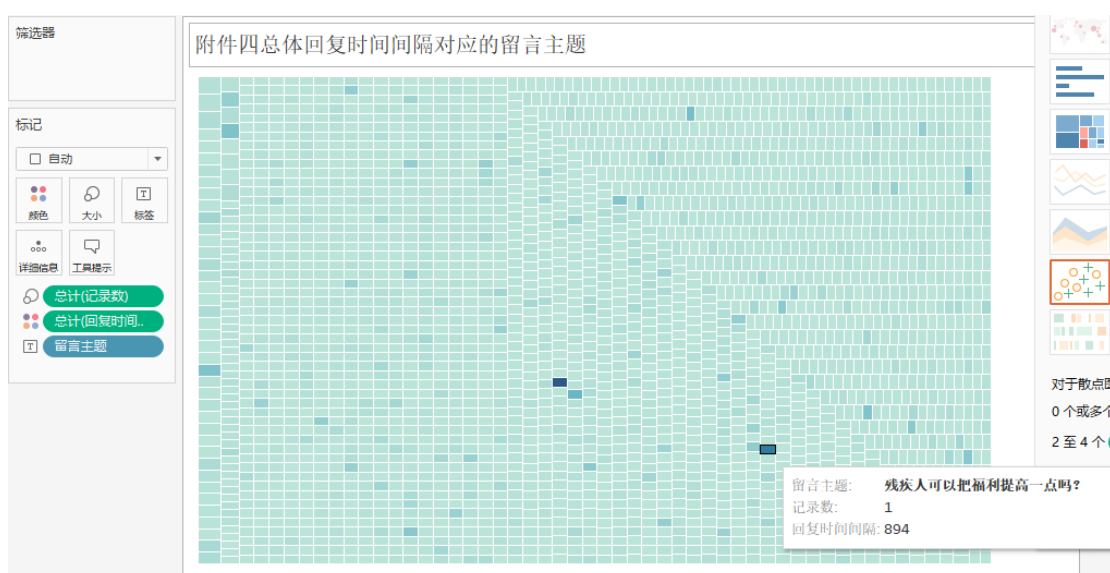
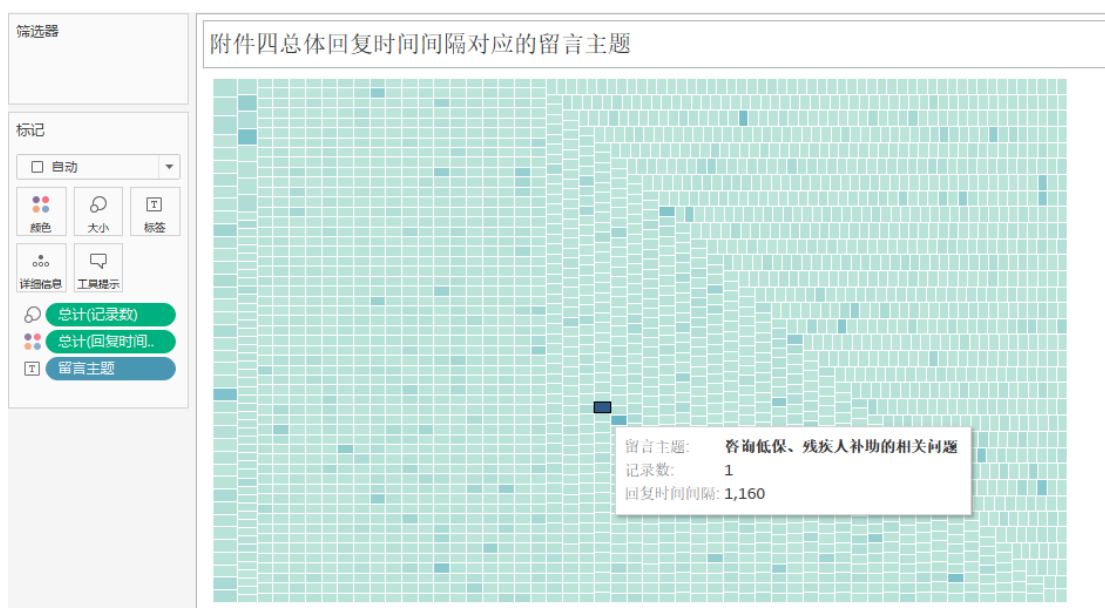
标记
自动
颜色
大小
文本
详细信息
工具提示
度量值
总计(留言编号)
总计(记录数)

所在地区回复时间间隔可视化

所在地区	回复时间间隔	留言编号	记录数
A1县	12.86722222	3,995	1
	14.926562500	4,452	1
	17.295277778	8,043	1
	21.001064815	8,448	1
	24.964768519	8,954	1
	28.874236111	4,089	1
	28.913078704	7,202	1
A2县	8.836886574	383	1
	11.229351852	965	1
	13.188460648	777	1
	15.225509259	2,549	1
	16.114074074	4,325	1
	17.603032407	4,218	1
	19.196238426	5,763	1
	19.271712963	5,283	1
	19.938726852	8,046	1
	23.854097222	4,032	1
	24.823506944	9,705	1
	37.311527778	9,930	1
	53.540266204	12,614	1
	70.733368056	3,685	1
A3县	4.028287037	6,453	1

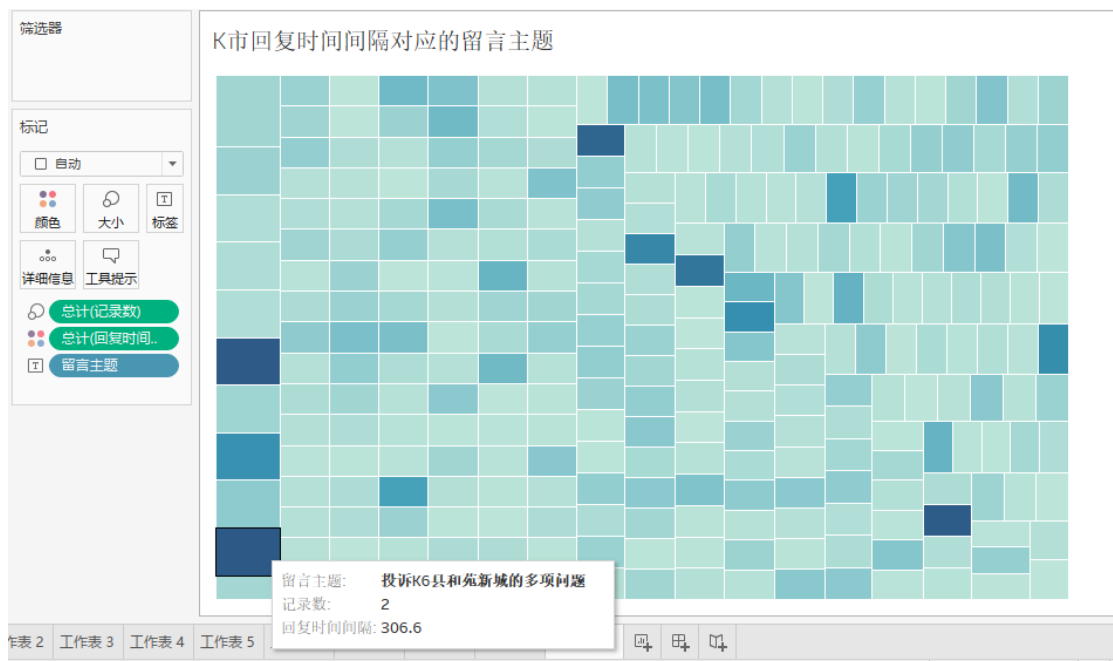
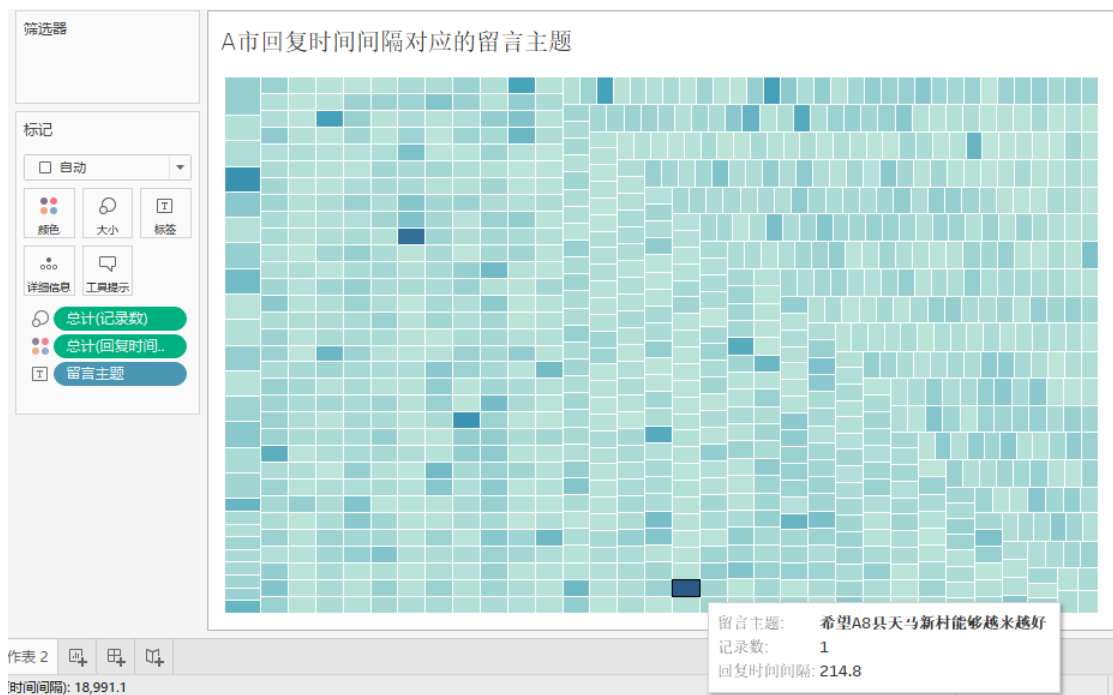
回复时间间隔: 28.874236111
所在地区: A1县
留言编号: 4,089

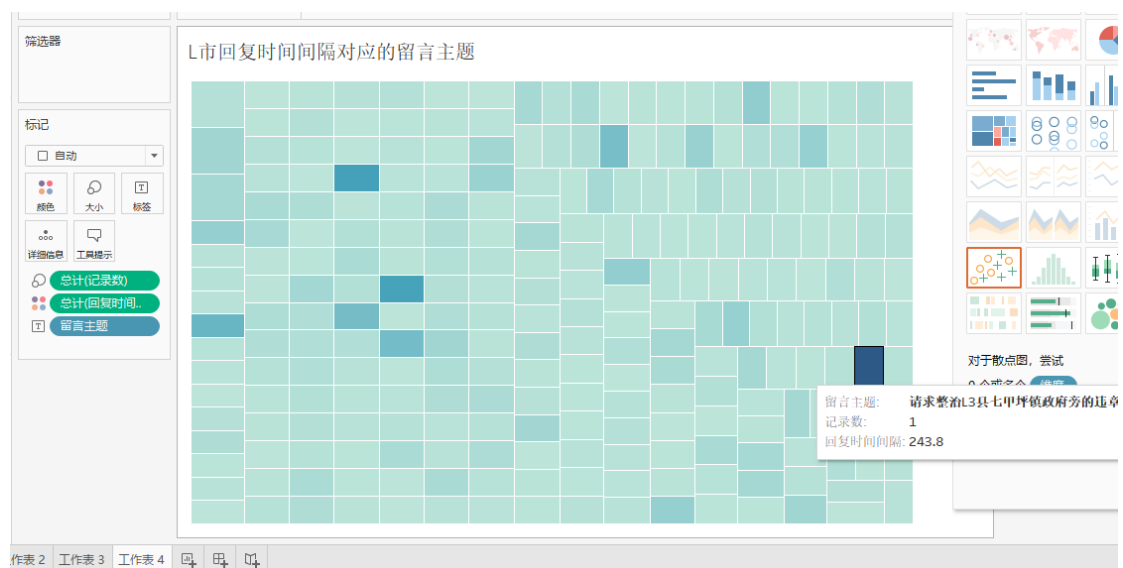
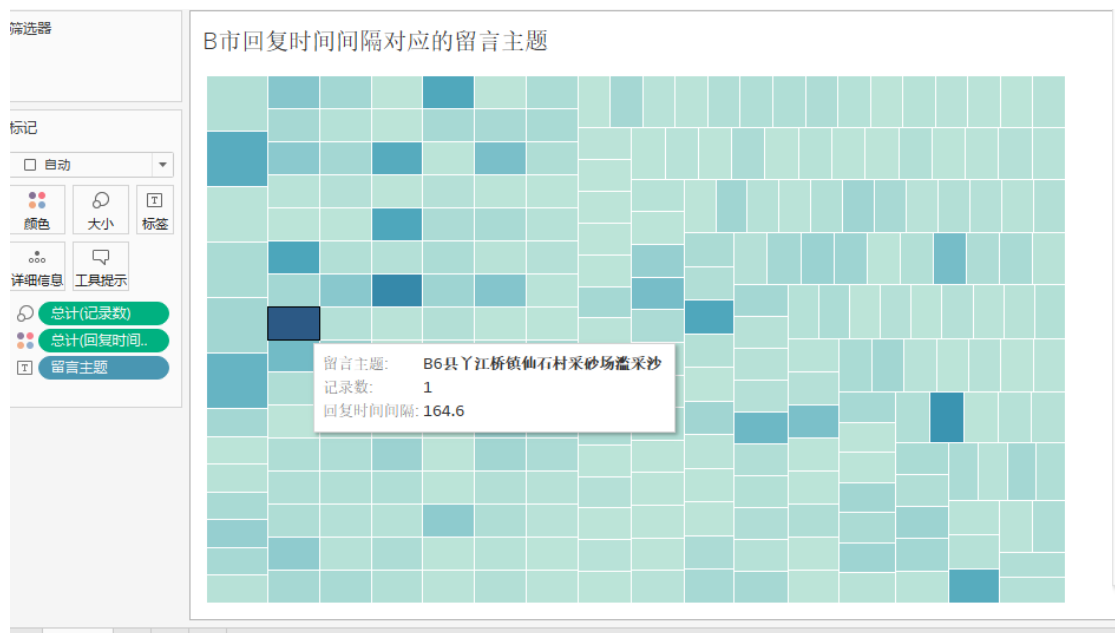
由上图，我们可以知道在气泡图显示的记录中，A 市存在较多的问题。

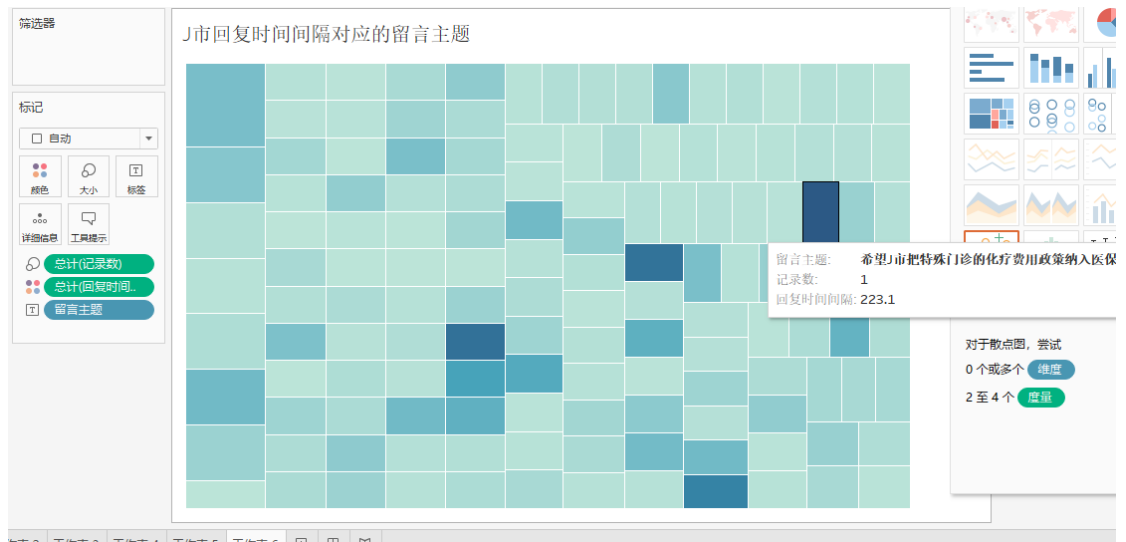
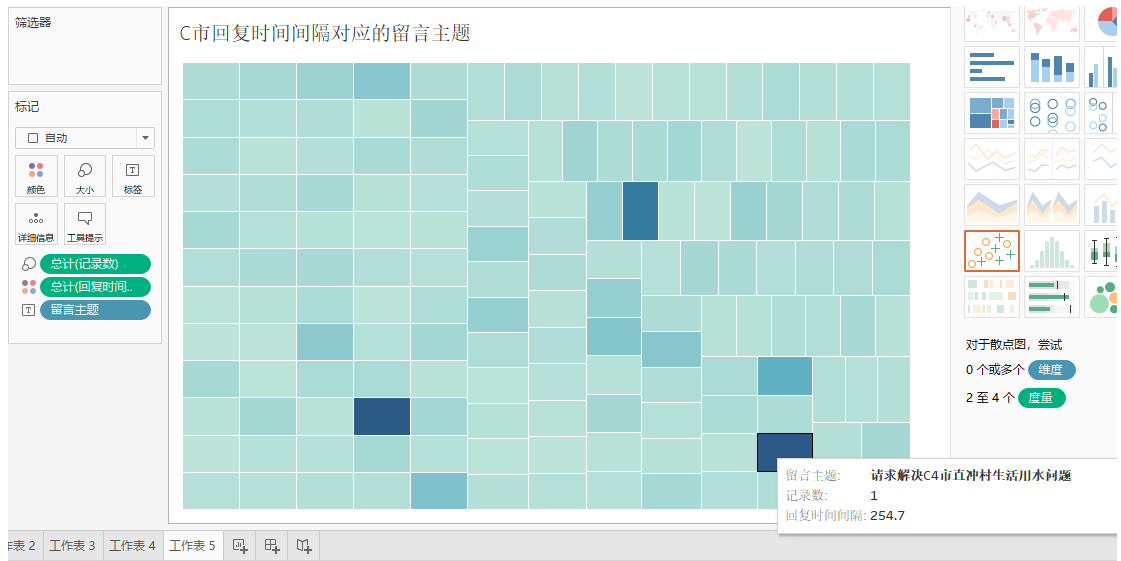


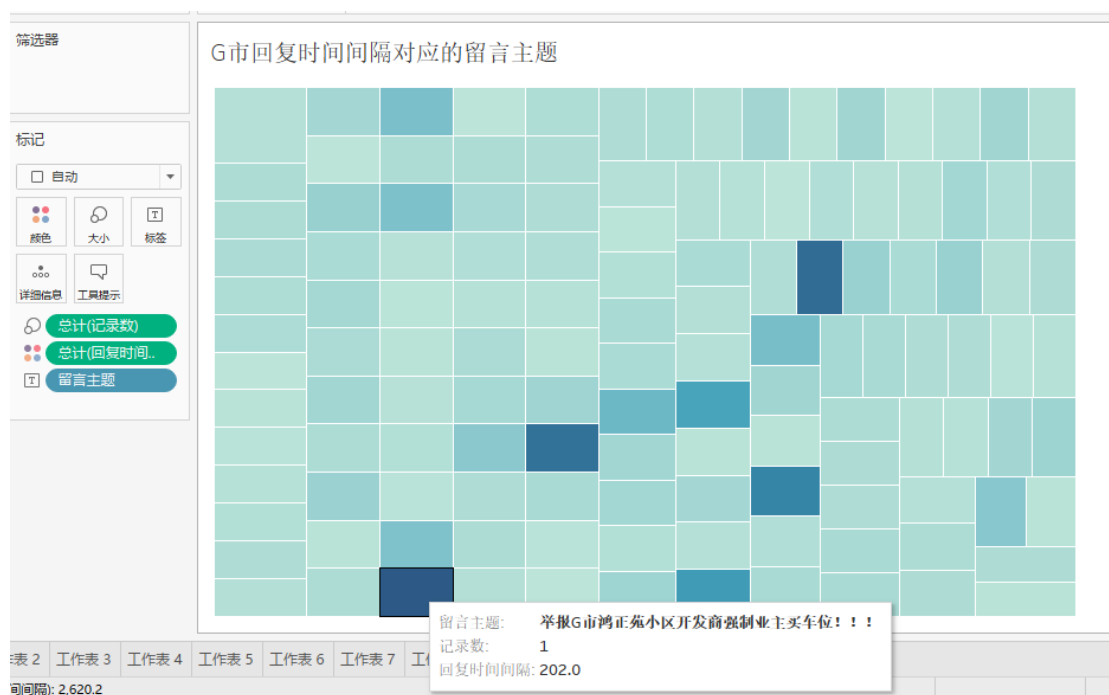
在附件四的所有记录中，我们发现在有关残疾人福利和低保也就是国家扶贫方面，答复时间间隔长达 1160 天，说明其处理程度偏难以及处理效率低。其次为生态安葬方面，两者均属第三标签，可见第三标签方面的留言较杂，同时答复时间偏长。

我们再来分析记录较多的 A 市、K 市、B 市、L 市、C 市、J 市以及 G 市。









图中,我们不难发现,颜色较深的方框表示回复时间间隔为最长,而数据可视化让我们更容易掌握数据的走向,从而能有效地解决更多问题。

由以上几组图片中,我们可以知道每个市的最长的回复时间间隔存在差异,而其中最长为 **301** 天,为 **K** 市的物业管理方面的处理。最短的为 **B** 市的生态环境问题的处理,为 **164** 天。从图中可知对城乡建设、物业管理、生态及医疗方面的处理所需要的时间较长,同时这些方面又与我们的生活息息相关,可以从侧面看出民生方面会存在较多的问题,而在对所在地区以及回复时间间隔长短可以侧面评估其答复的相关性、完整性以及可解释性。

4.结论

网络问政对政府及时了解民意、解决民生问题有重要作用,因为

各类社情民意相关的文本数据量不断攀升，对机械学习，文本分析等现代技术的一大难题，。本文采用根据 jieba 分词，构造词向量模型，运用 TF-IDF 算法，贝叶斯分类等对留言信息进行分类分析等操作，统计了群众留言的类型，并努力定义出民生最关注的热点问题，分析了问题的走向和对政府相关部门给出的回复做出评价方案。

由分析结果可以看出，道路的拆迁、公交建设，询问对闲置场地的规划，违建之风盛行，拖欠工人工资进半年，土地闲置，设施建设差等问题是群众迫切想要了解和解决的问题。对群众留言问题的分析，有利于快速定位政府相关部门的工作方向，对群众做出回复，解决民生问题。

5.参考文献

[1]特征向量法建模机理 作者：黄德 刊名：数学的实践与认识 上传者：陈欣柏

[2] 网络涉警舆情热度评价指标体系构建 孙吟龙 1 周捷 2

[3] 基于新浪热门平台的微博热度评价指标体系实证研究

梁昌明 1 (1. 山东师范大学历史与社会发展学院，济南

李冬强 2 250014； 2. 北京科技大学图书馆，北京 100083)