“智慧政务”中的文本挖掘应用

摘要

近年来，随着云计算、物联网、互联网等技术的蓬勃发展，由“互联网+政务服务”构建的智慧型政府——“智慧政务”应运而生。其运用互联网、大数据等现代信息技术，简化群众办事环节、提升政府行政效能。目前微信、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。各类社情民意相关的文本数据量的不断攀升，给依靠人工来进行留言划分和热点整理的相关部门的工作带来了极大挑战。本文以此情况为出发点，基于自然语言处理技术，结合 TF-ITF 算法、多项式朴素贝叶斯算法、K 均值聚类算法等方法，对群众留言进行文本挖掘分析，制定合适的分类体系、热度评价指标、回复内容评价指标，以期解决群众留言分类、热点问题挖掘和答复意见评价等问题。

针对问题 1，本文利用 Python 编程语言 pandas 库中的 drop_duplicates 函数对附件 2 的留言详情进行去重处理，然后利用 jieba 库，对留言进行分词操作，划分测试集与训练集。基于 TF-IDF 策略，获取训练文本和测试文本的 TF-ITF 权值向量，将文本转换为向量表示。继而利用朴素贝叶斯算法，构建分类模型，最后对模型进行训练并采用 F-Score 对模型进行评价，查看效果，对模型进行改进，绘制混淆矩阵，进行模型结果可视化。

针对问题 2，本文观察附件 3 数据以及热点问题示例表，利用 pandas 对挖掘数据进行清洗。由于总体数据量较大，本文以欠抽样的方式提取部分数据进行文本热点挖掘实验，首先对文本数据进行文本特征项提取，接着利用 TF-IDF 算法将文本向量化，再采用相似度公式（余弦公式）进行相似度计算和对比，最后采用 K 均值聚类算法进行聚类。分析每个类别，针对居民关注的热点问题得出一个留言热度计算的公式，进行相应的热度指数计算，然后进行排序，得到热点问题表。

针对问题 3，本文运用计算文本相似度、构建话题模型等方法，结合留言人心理等因素从答复的相关性、完整性、可解释性等角度对答复意见的质量给出了一套评价方案。

关键词：中文分词；TF-IDF 算法；多项式朴素贝叶斯算法；K 均值；留言评价；话题模型

Abstract

In recent years, with the vigorous development of cloud computing, Internet of things, Internet and other technologies, the "smart government" emerged as the "intelligent government", which is built by the Internet plus government service. It uses the Internet, big data and other modern information technology to simplify the mass work and improve the administrative efficiency of the government. At present, wechat, Weibo, mayor's mailbox, sunshine hotline and other online political platforms have gradually become an important channel for the government to understand public opinion, gather people's wisdom and gather people's spirit. The increasing amount of text data related to social situation and public opinion has brought great challenges to the work of relevant departments which rely on manual message division and hot spot sorting. Based on this situation and natural language processing technology, combined with tf-itf algorithm, polynomial naive Bayes algorithm, K-means clustering algorithm and other methods, this paper analyzes the text mining of mass message, and develops appropriate classification system, heat evaluation index, response content evaluation index, so as to solve the problems of mass message classification, hot issue mining and response opinion evaluation Question.

To solve the problem 1, this paper uses the drop - duplicates function in the python programming language pandas library to de duplicate the message details in attachment 2, and then uses the Jieba library to segment the message, divide the test set and training set. Based on TF-IDF strategy, tf-itf weight vectors of training text and test text are obtained and transformed into vector representation. Then we use naive Bayes algorithm to build the classification model. Finally, we train the model and use F-score to evaluate the model, check the effect, improve the model, draw the confusion matrix, and visualize the model results.

For problem 2, this paper observes the data in attachment 3 and the sample table of hot issues, and uses pandas to clean the mining data. Because of the large amount of total data, this paper extracts part of the data in the way of under sampling for text hot spot mining experiment. Firstly, text features are extracted from the text data, then TF-IDF algorithm is used to quantify the text, then similarity formula (cosine formula) is used for similarity calculation and comparison, and finally K-means clustering algorithm is used for clustering. Analysis of each category, according to the hot issues concerned by the residents, we get a formula for calculating the heat of messages, and then calculate the corresponding heat index, and then sort it to get the hot issues table.

In view of question 3, this paper uses the methods of calculating text similarity,

constructing topic model, and combining with the factors such as the psychology of the commenter to give a set of evaluation scheme for the quality of the reply from the perspective of relevance, integrity, and interpretability.

Keywords: Chinese word segmentation; TF-IDF algorithm; polynomial naive Bayesian algorithm; k-means; comment evaluation; topic model

目录

1.挖掘目标.....5
2.分析方法与过程.....5
2.1 问题一 分析方法与过程.....5
2.1.1 流程图.....5
2.1.2 数据预处理.....5
2.1.3 TF-IDF 策略.....8
2.1.4 ComplementNB()分类器.....8
2.1.5 F-Score 模型评价.....9
2.2.5 模型可视化.....9
2.2.6 模型拓展.....10
2.2 问题二 解决方法与过程.....11
2.2.1 问题分析及流程图.....11
2.2.2 数据预处理.....11
2.2.3 文本数据特征提取.....11
2.2.4 文本向量化.....12
2.2.5 相似度计算.....12
2.2.6 K 均值聚类.....13
2.2.7 热度指标.....15
2.2.7 热点问题表.....16
2.3 问题三 分析方法与过程.....18
2.3.1 目标概述.....18
2.3.2 居民心理分析.....18
2.3.3 回复内容评价指标.....18
2.3.4 各类指标计算方案.....18
参考文献.....20

1. 挖掘目标

本文旨在利用基于自然语言处理技术的智慧政务系统助力政府相关部门对居民留言意见的处理。利用中文分词、TF-ITF 权值策略，朴素贝叶斯算法，word2vec、K 均值、余弦公式等完成下面三个问题：

①对大量居民留言就行分类处理，将各种留言划分到相应的一级分类目录下。

②将某一时段内反映特定地点或特定人群问题的留言进行归类，定义合理的热度评价指标，得到排名前五的热点问题，并以表格形式展现。

③分析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，设计出一套留言评价方案。

2. 分析方法与过程

2.1 问题一 分析方法与过程

2.1.1 流程图

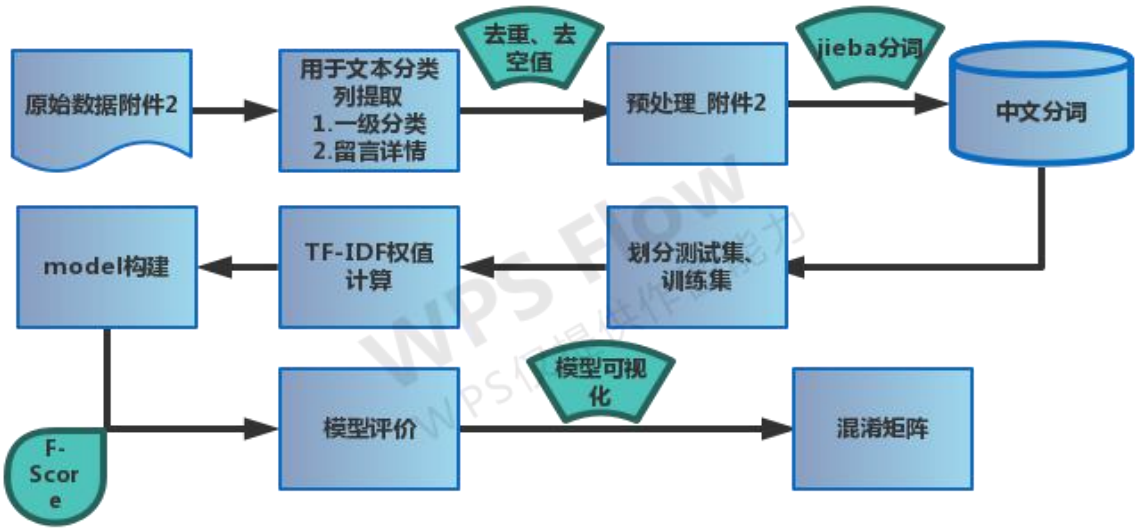

图 2-1 问题 1 流程图

2.1.2 数据预处理

1. 读取数据（数据提取有用列）

①分析附件 1 中的“一级分类”列，观察各类一级标签所占比例，运用 python 编程语言中 pyecharts 库做可视化分析得到一级标签圆环图，如图 2-2 所示。从图中可以看出，一级标签“城乡建设” 包含的下级标签最多，一级标签“国土资源”包含的下级标签最少。

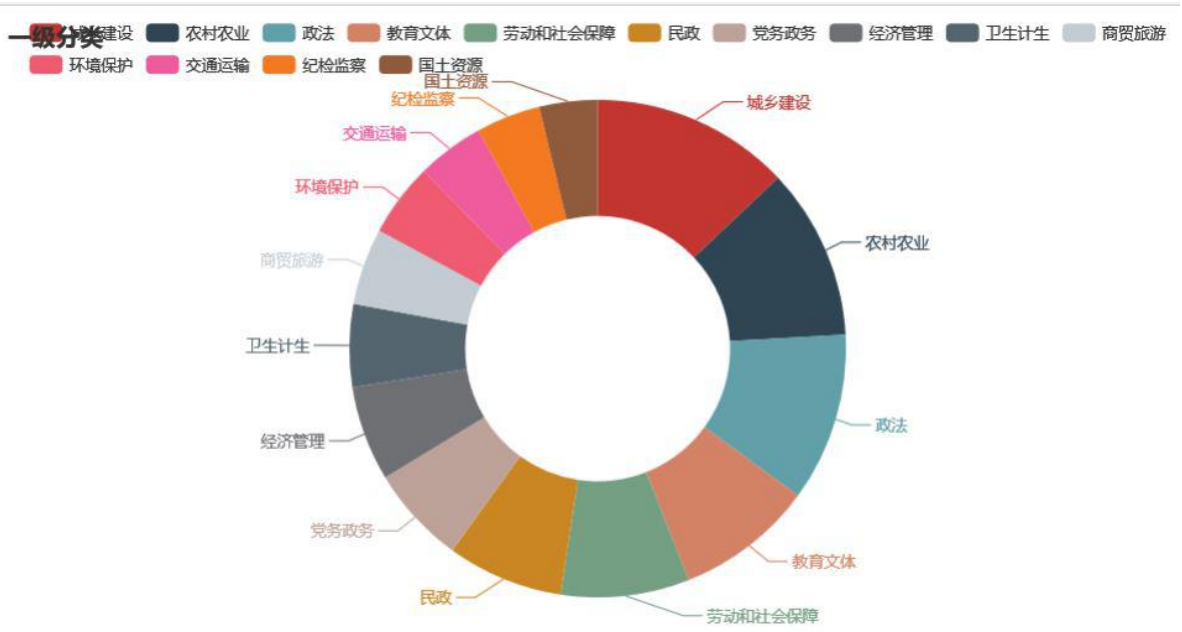

图 2-2 一级标签圆环图

②分析附件 2 的数据，提取其中“留言详情”和“一级分类”两列进行分析处理，运用上节所述方法对附件 2 中的“一级分类”列做可视化分析得到留言的一级分类圆环图，可以清晰地看到各类留言的比例。继而通过对附件 2 的“一级分类”列进行排序统计，得到每一类的具体占比值，其中劳动和社会保障类的留言占 21.38%，城乡建设类的留言占 21.81%，交通运输类占比最少，仅占留言的 6.66%。

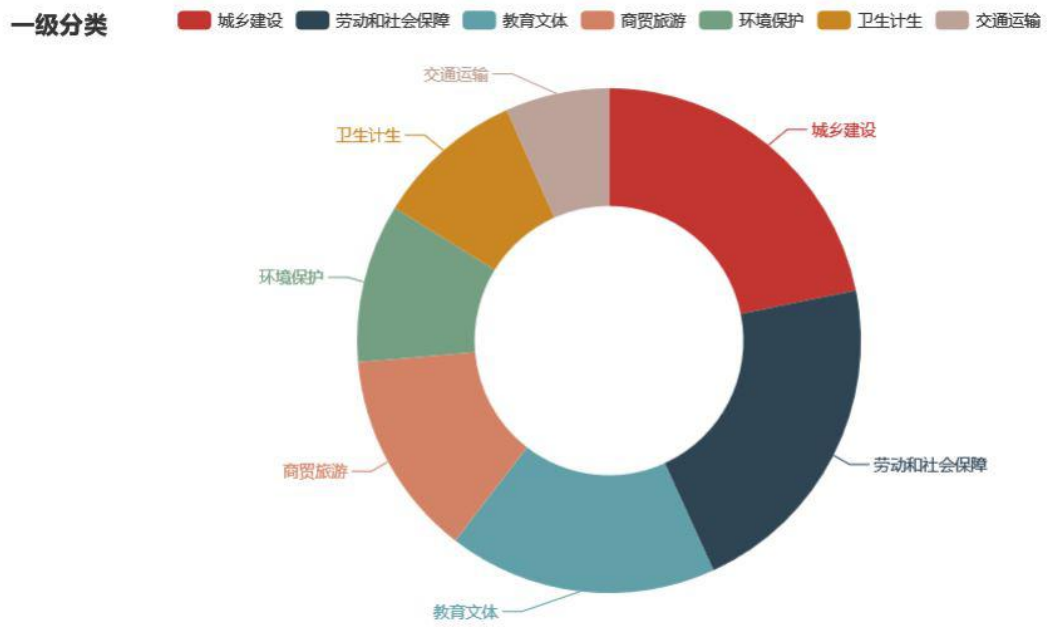

图 2-3 留言详情一级分类圆环图

2.数据清洗（去重、去空、去异常值、去换行符）

①对在读取数据步骤中构造的只有“留言详情”和“一级分类”的新数据进行去重操作。

②对在读取数据步骤中构造的只有“留言详情”和“一级分类”的新数据进行去空操作。

③在读取的数据中，某些留言的字数非常多，而且里面含有换行符、制表符、空白符等，故本文使用正则表达式替换该符为空。

清洗原则：由于解决问题一只会用到附件 2.xlsx 中的两列数据，所以在尽量保留多的数据的情况下，本文只针对分类列和留言列进行重复、空值判断，重复行保留第一项数据，空值行会很大程度的影响后续操作，故直接删除。

④为了减少计算量，删减冗余，本文利用机械压索去词的方法，将单一重复和过分强调 3~5 次及以上的频次的词删减掉，并将无实际意义的短句进行删减，最后将清洗后的数据保存到附件 2_清洗.csv.

3. 中文分词

因为留言数据量多，且留言内容结构混乱，所以在对留言文本进行文本向量化及模型构建之前，需要将这些文本数据转换为计算机能够识别的结构化数据形式。

①将处理好的数据利用 jieba 中文分词工具，导入停用词表，并自定义停用符，对数据进行分词操作，得到分词效果，如下表所示。

表 2-1 分词效果表

序号	分词效果									
6378	市市	企	青峰	煤矿	一名	正式	职员	六级	残疾	军人
6786	F	市	邮政储蓄	银行	巴陵	中路	收取	社会保险	卡	短信
6942	I4	县	通锑业	有限责任	公司	原	职工	五人	实名	举报
5777	请求	政府	清算	烂尾	工程	抓捕	诈骗	嫌疑人	投诉	书
2054	经阁	铝材	先导	区	污染	大户	河西	搬迁	年	河西
2238	市	邓	环保局	废塑料	本市	市民	市里	环保局	局长	大开
2352	桃园路	14	发射塔	中国移动	建立	信号	居民	严重危害	号	楼上
2068	尊敬	领导	市	大道	蓝色	港湾	小区	住户	蓝色	售
2786	刘尧臣	厅长	您好	B	市	焚烧	垃圾	发电厂	运行	项目

②经过附件 2 数据的分析，可以得到在所有留言中城乡建设类的留言条数最多，下图是其词云图，可以看出“市”、“业主”是该分类留言下的 TF 值最大的字。

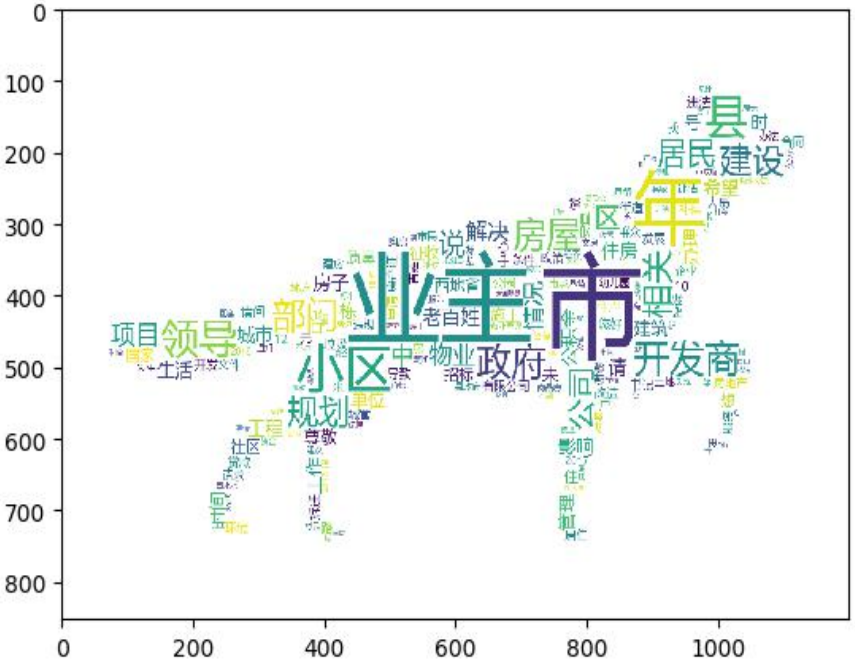

图 2-4 城乡建设词云图

2.1.3 TF-IDF 策略

①TF-IDF 原理及使用

TF-IDF：词频逆文档权重方案。

原理:如果某个词或者短语在一篇文章中出现频率高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类。

TF(词频):指的是某一个给定词语在文件中出现的频率。

First:TF = 该词在文件中出现的次数/在文件中所有字词的次数之和

IDF(逆向文本频率):是一个词语普遍重要性的度量。

Second:IDF = log(总文件数目/(包含该词语的文件的数目+1))

考虑到在计算过程中可能出现分母为 0 的情况，分母为 0 等价于所有留言中都未出现该词，分母位置加上一个 1，防止出现分母为 0 的情况。

计算 TF 和 IDF 的乘积，值越大说明该词越重要。

Third:TF-IDF = TF(词频)×IDF(逆向文本频率)

2.1.4 ComplementNB()分类器

模型原理：

①新建 py 脚本，导入预先准备好的数据。

②将分词后的数据按 8:2 的比例划分测试集和训练集合，并保持测试集和训

练集样本的维度一致。

③分别获取训练样本和测试样本的 tf-idf 权值：

TF-IDF = TF（词频） × IDF（逆文档率）

④构建模型：在生成 TF-IDF 权值向量后，我们开始构建分类模型，查阅资料及实际操作，发现多项式朴素贝叶斯在该分类中的效果最好。

2.1.5 F-Score 模型评价

F-score 计算方法：

即对模型的精确率和召回率进行加权平均

F-score=(2*precision*recall)/(precision+recall)

对多项式朴素贝叶斯分类模型进行评价得到结果如下：

表 2-2 多项式朴素贝叶斯分类模型评分表

模型名称	TESTING F1 SCORE
多项式朴素贝叶斯模型	0.8929359823399559

模型评价：对该模型进行多次 F-score 评分，平均分为 0.89，说明该模型性能较好，在实际应用中可以取得良好的效果。

2.2.5 模型可视化

原理:混淆矩阵是一种特殊类型的列联表(contingency table)形式或者交叉制表(cross tabulation or crosstab)形式，其有两个维度(真实值 “actual” 和预测值 “predicted”)；这两个维度都是具有相同的类(“classes”)的集合在列联表中，每一个维度和类的组合是某一个变量通过表的形式,可视化地将多个变量的频率分布展示出来。

绘制混淆矩阵：

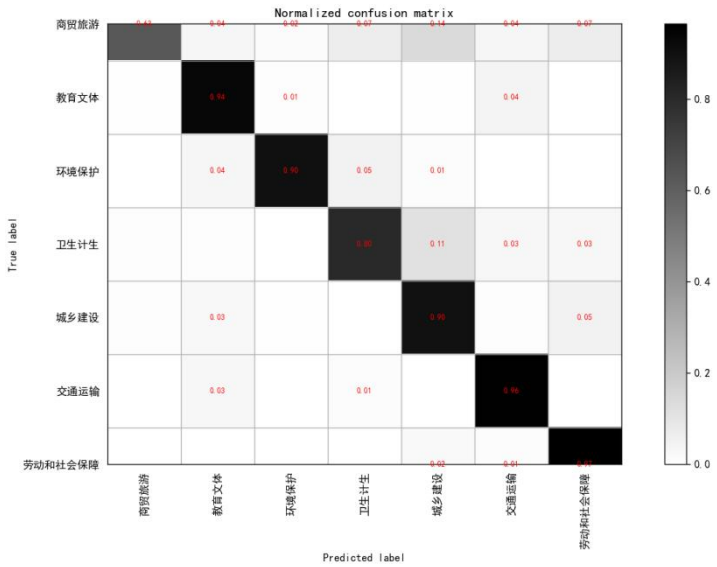

图 2-5 分类模型可视化

2.2.6 模型拓展

在查阅资料后我们利用我们利用文档的向量化表示方法 Doc2Vec，来建立分类模型，并且使用 F-Score 对分类方法进行评价。

新建 question1.py 脚本文件来实现该模型

第一步：利用我们将首先使用 Doc2vec 的 DBOW 算法，我们通过训练神经网络来获得相应的文档向量（而不是使用传统计算 IF-IDF 权值方法），该神经网络将用于预测段落中的单词的概率分布，给出来自段落的随机采样的单词，在此基础上创建 dbow 分类模型。

第二步：使用 Doc2vec 的 DM 算法，设置 dm 的参数为 1，创建 dm 的模型。

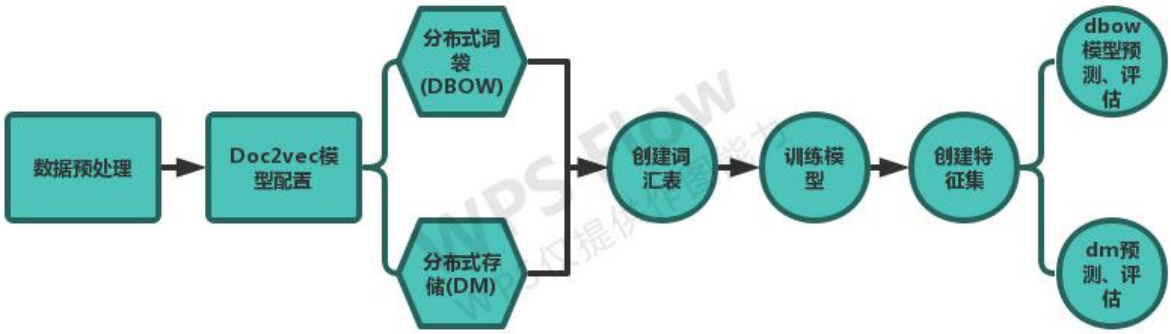

图 2-6 Doc2Vec 文本分类模型流程图

接下我们分别对这两类模型进行 F-Score 评分,得到模型评分表，如下表 2-2 所示。由表可知，两类模型的评分均在 0.7 以下。

表 2-2 模型评分表

模型名称	TESTING ACCURACY	TESTING F1 SCORE
Dbow 模型	0.6644295302013423	0.6525341702695454
Dm 模型	0.5906040268456376	0.5715691790611171

第三步：合成一个新模型

查阅资料，将分布式存储(DM)和分布式存储(DM)合成为一个新模型,然后再用这个新模型来创建文档特征向量，可以一定程度上提高模型的性能。

利用 F-Score 对合成的新模型进行打分，得到新模型评分表，如下表 2-3 所示。新模型的评分为 0.6848635207893362，在前两个模型的基础上确实有所提高，但是在该问题上，其性能不及多项式朴素贝叶斯模型。

表 2-3 新模型评分表

模型名称	TESTING ACCURACY	TESTING F1 SCORE
Dbow 模型	0.6644295302013423	0.6525341702695454
Dm 模型	0.5906040268456376	0.5715691790611171
新模型	0.697986577181208	0.6848635207893362

2.2 问题二 解决方法与过程

2.2.1 问题分析及流程图

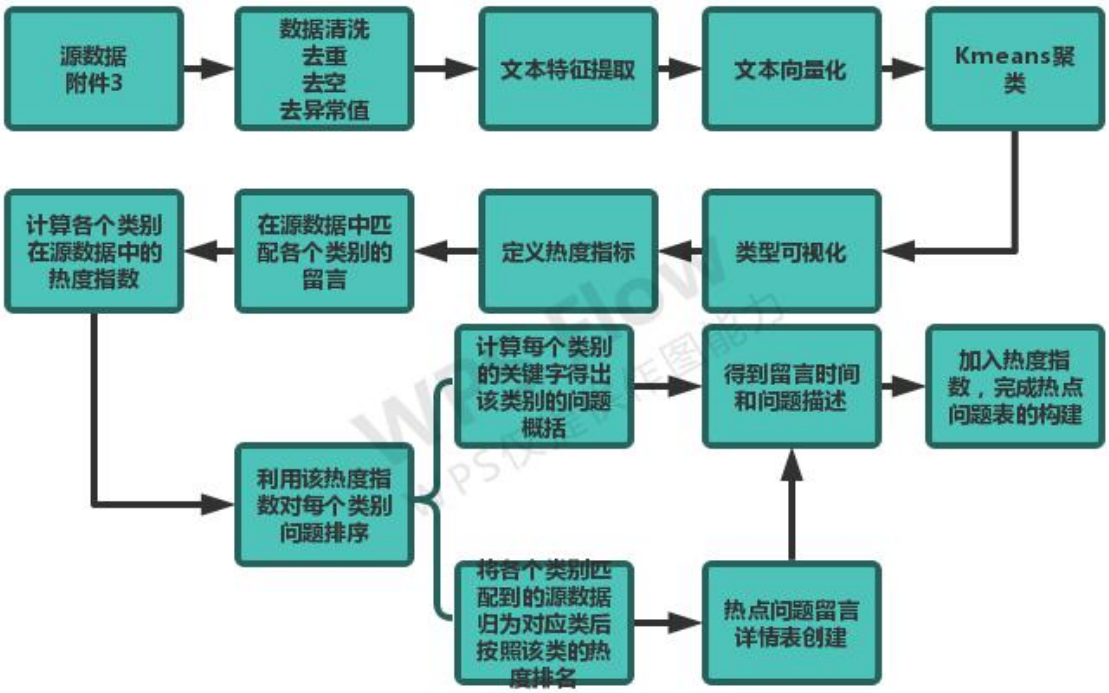

图 2-7 问题 2 流程图

2.2.2 数据预处理

对附件 3 文本数据进行去重、去空值、去异常值处理，为了尽量不丢失数据，本文将保留第一项重复数据，直接删除空值和异常值。

2.2.3 文本数据特征提取

提取附件 3 中“留言主题”列和“留言时间”列，利用 jieba 分词后对留言主题进行分词操作和统计词频，并绘制词云图，如下图所示。从词云图中，我们可以看见词频最大为“A 区”，结合源数据可以初步得出居民留言问题基本集中在

A 区。

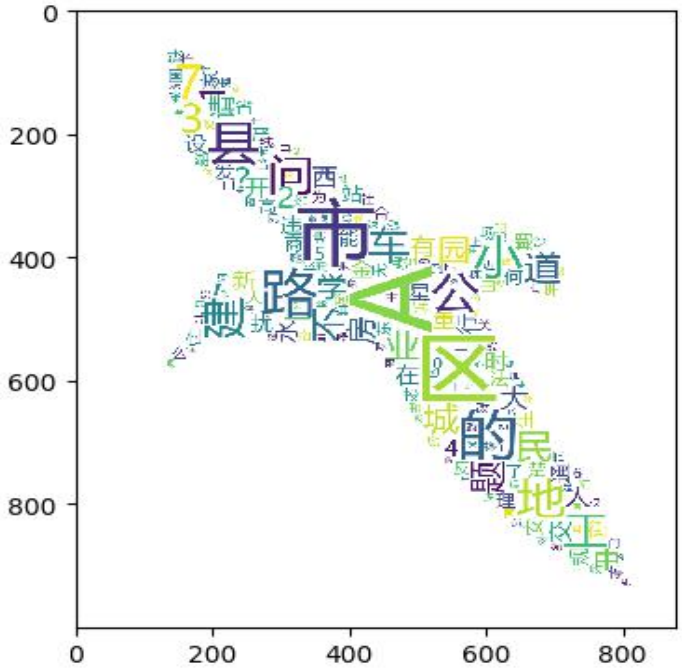

图 2-8 留言主题词云

2.2.4 文本向量化

我们发现留言中有许多地理名词（如：A 市、A3 区...），但 jieba 库分词将不会区分该类地点名词，所有我们新建“专用词.txt”文档，向 jieba 库中导入相应的专用词.txt，然后对留言主题进行 jieba 分词，再将分词列表集转换成稀疏向量集，建立相应的语料库。

我们导入 CountVectorizer、TfidfTransformer 库，对语料库进行 TF-IDF 权值的计算，得到对应的稀疏矩阵。

2.2.5 相似度计算

得到每个留言详情的 tf-idf 权值稀疏矩阵后，我们利用计算两个向量之间的余弦值来获得每两条留言主题之间的相似度，如图 2-9 所示。

自定义函数完成相似度的计算。

假设文本 A 是个 n（a1,a2,...,an）维向量，B 也是 n(b1,b2,b3,...,bn)维向量，则 A,B 之间的余弦值计算公式：

$$\cos\theta=\frac{\sum_{i=1}^{n}(Ai\times Bi)}{\sqrt{\sum_{i=1}^{n}(Ai)^2}\times\sqrt{\sum_{i=1}^{n}(Bi)^2}}=\frac{A\cdot B}{|A|\times|B|}$$

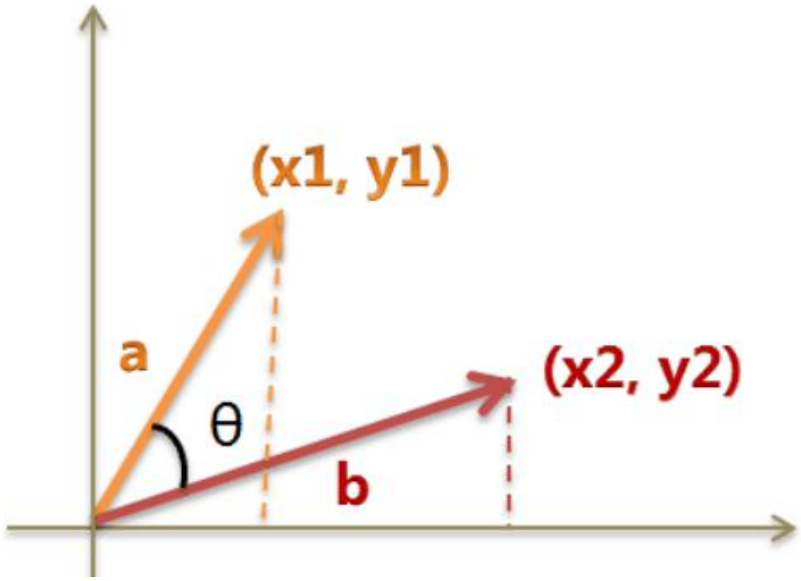

图 2-9 余弦定理

2.2.6 K 均值聚类

k 均值聚类算法（k-means clustering algorithm）

基本原理：

①预将数据分为 K 组，则随机选取 K 个对象作为初始的聚类中心。

②然后计算每个对象与各个种子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心。

基本步骤：

假设我们输入的样本为 a=a1,a2,a3,...,an.

第一步：选择 K 个中心:b1,b2,b3...bk.

第二步：对于每个输入的样本 ai,将该样本划分给聚类分类中心最近的那一类：

$$\text{label}_i = \arg\min_{1\le j\le k} \|a_i - b_j\|$$

第三步:将每个类中心更新为类别中所有当前样本的均值：

$$b_j = \frac{1}{|c_j|}\sum_{i\in c_j} \text{a}_i$$

重复第二、三步骤，使得类别中心小于预定阈值。

在该问题中，结合题目中给出的热点问题表，我们利用 K 均值聚类，得到聚类中心，预先设置类别参数为 30，分析聚类结果，我们将每个类别进行留言条数的统计。

利用 pyecharts 对留言条数进行可视化操作，得到图 2-10.

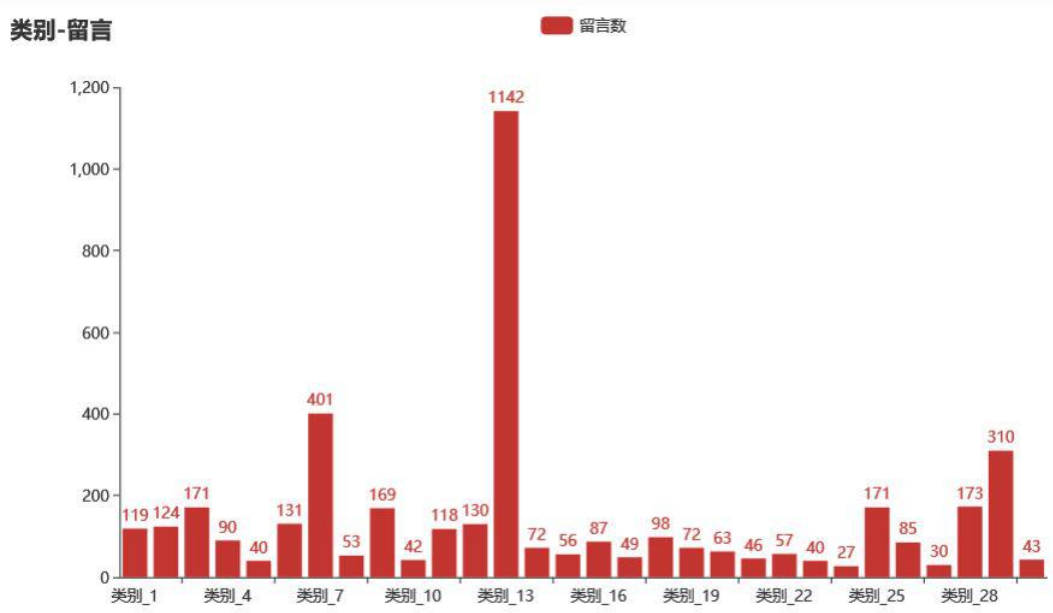

图 2-10 类别留言数可视化图

聚类结果可视化：

下图为我们利用 T-SNE 算法对聚类结果降维后，使用 matplotlib 对结果进行的可视化操作。

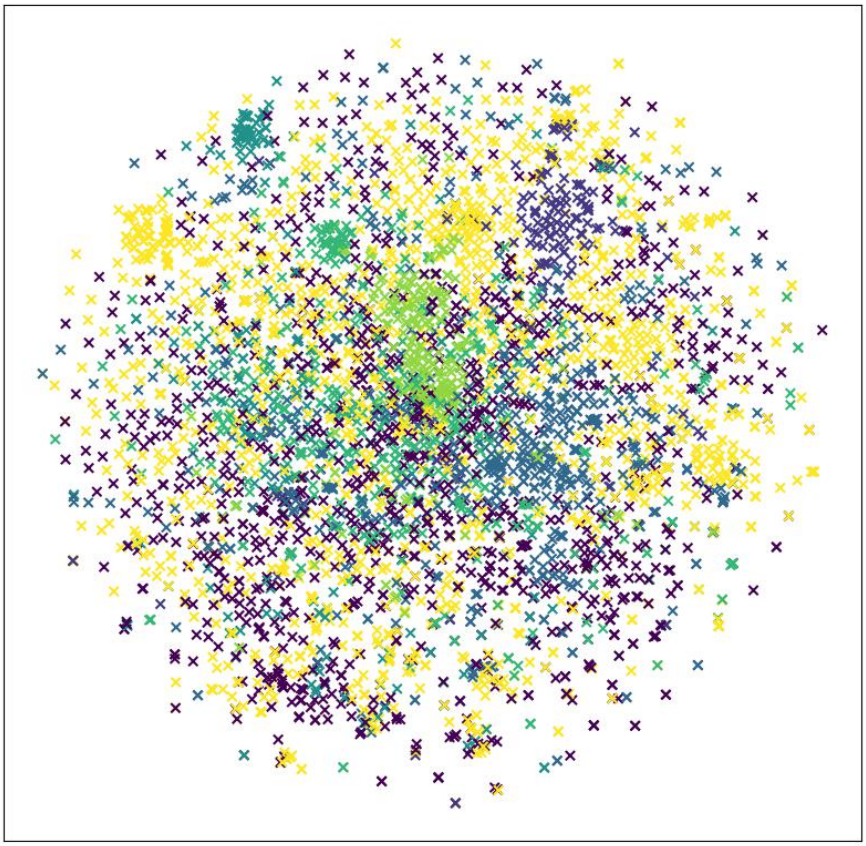

图 2-10 聚类分析

基于 K 均值聚类后的各类留言中，部分类别聚类效果显著。在下面的聚类信息表中，绝大多数 ID 中都出现了“丽发”、“新城”、“搅拌站”、“噪音”、“扰民”等词语，其表达的语义相近，由此可以看出该类别分类效果显著。

表 2-4 聚类信息表

ID	聚类中心
33	A 市 万家 丽 南路 丽发 新城 居民区 搅拌站 扰民
68	投诉 A2 区 丽发 新城 建 搅拌站 噪音 扰民
77	丽发 新城 小区 旁边 建 搅挥站
98	A 市 丽发 新城 违建 搅拌站 施工 扰 民污染环境
105	A 市 丽发 小区 建 搅拌站 噪音 污染
153	A 市 A2 区 丽发 新城 道路 坑坑洼洼
463	A2 区 丽发 新城 修建 搅拌厂 污染环境
632	A 市 丽发 新城 小区 侧面 建设 混泥土 搅拌站 粉尘 噪音 污染
832	投诉 小区 搅拌站 噪音 扰民
848	A2 区 丽发 新城 修建 搅拌站 污染环境 影响 生活
1050	投诉 丽发 新城 小区 违建 搅拌站 噪音 扰民
1072	A2 区 丽发 新城 违规 乱建 混凝土 搅拌站 监管
1089	A 市 丽发 新城 小区 搅拌站 噪音 扰民 污染环境
1153	A2 区 丽发 新城 小区 旁边 搅拌厂 合法经营
1165	A2 区 丽发 新城 小区 太吵

2.2.7 热度指标

在获得聚类后的各个类别数据，我们分别观察每个类别中的文本数据，看是否符合要求，反复对比后，我们分析分类数是符合要求的，并能实现 K 均值的聚类计算。

为了便于后面热点问题表的制作，我们将每个类别的数据利用 DataFrame 进行格式化后输出为 csv 文件。

表 2-5 TF-IDF 权重表

矩阵坐标	TF-IDF 权重
（0，141）	0.2998353219018954

（0，142）	0.12640094922201325
（0，261）	0.4241576090590338
（0，231）	0.46051954705444054
（0，122）	0.3619964654804646
（0，87）	0.4241576090590338
（0，265）	0.3241576090590338
（0，7）	0.10617369527409341

在表 2-5 中，矩阵中的坐标（0，141），表示第 141 个关键词在第 0 个句子中的 TF-IDF 权重，得到这个规律后，开始构建热度指数。

热度排名是基于所有留言的排序，将每个类别的各个留言在结果处理的附件 3_1.xlsx 中匹配到原数据，计算每个条留言的在所有中的平均 tf-idf 权值，再求出每个类别的平均权值。

每个类别的平均 TF-IDF = sum（该类别中每条留言的平均 td-idf）/该类别中的留言条数。

这里本文将每个类别在所有类别中的平均 tf-idf 值来作为热度指标。

2.2.7 热点问题表

导入一个如表 2-4 的类别 csv 文件。

第一步：构建热点问题表

在初步对附件 3.xlsx 进行 K 均值分类后，进一步分析每一类的数据，对分类效果好的类别进行关键词的 TF-IDF 计算、排序，进行该类别留言问题的概括；对第一步 K 均值分类效果不明显的数据进行进一步的 K 均值分类，重复上步操作，直到每一类中基本不含有其他语义的留言。

①分析每个类别，自定义函数 Key_word_TF_IDF(),调用该函数，统计每个类别的关键词，计算每个关键词的 tf-idf 权值；从权值最大的几类词语中提取可以描述该类别的的关键句，并提取出相应位置信息。

利用 for 循环，循环调用该函数，完成所有类别的关键词和 tf-idf 计算，并利用 pandas 中主键合并的方式合并关键词和 tf-idf 输出到对应的 csv 文件中。

②将每个类别的数据匹配到源数据附件 3.csv,将匹配到的数据保存为类别匹配数据.csv，代码详情见匹配源数据.py。

③在每个类别匹配数据中，提取时间列的年月日数据，进行时间顺序排列，得到该类别留言的时间范围。

④计算利用准备好的热度指标，计算每个类别的热度指数，并添加问题 ID。

⑤结合前四步，构建热点问题表，保存为热点问题表.xlsx.

第二步：构建热点问题留言明细表

结合第一步将每个类别匹配到 CSV 文件对应类别，赋值问题 ID，构建热点问题留言明细表，保存为热点问题留言明细表.xlsx。

表 2-6 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
Ⅰ	1	3.76229	2019/2/16-2019/11/26	A3 区	临街餐馆油烟噪音扰民
Ⅱ	2	3.7566	2019/1/2-2019/9/26	A3 区	占道经营、占道停车问题
Ⅲ	3	3.6628	2017/6/8-2020/1/6	西地省部分公司	偷税漏税、诈骗、拖欠工资等问题
Ⅳ	4	3.63302	2019/1/13-2020/1/6	A 市	房屋质量问题和违规搭建问题
Ⅴ	5	3.63232	2019/7/7-2019/8/26	伊景园滨河苑	捆绑销售车位问题

在表 2-6 中，排名前 2 的热点问题均发生在 A3 区，分别是“临街餐馆油烟噪音扰民问题”和“占道经营、占道停车问题”。将热点问题进行可视化操作，得到图 2-11，该图中为所有留言中居民集中反映的热点问题，包括占道、民生经济、捆绑销售车位、房屋质量及搭建等问题。

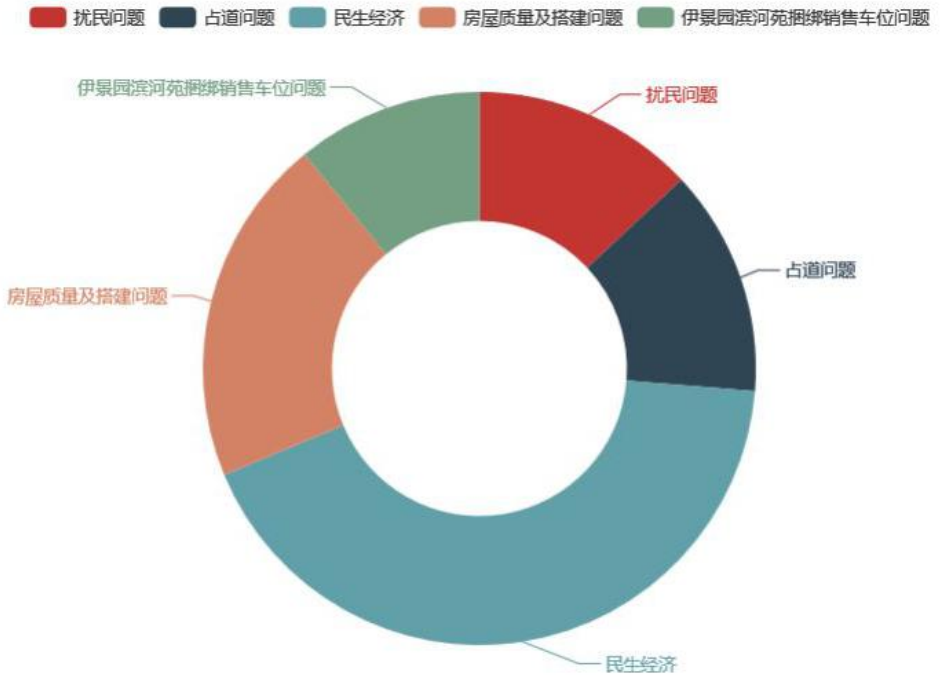

图 2-11 前五类热点问题圆环图

2.3 问题三 分析方法与过程

2.3.1 目标概述

针对居民的留言问题，是居民希望政府可以尽快出面处理的问题，而最先的一步则是在留言回复中给予居民答案。我们希望回复居民留言的内容是可以尽量科学的，完整的给居民想得到的结果。所以我们分析居民希望的到回复的各种心里状态，对每条留言回复给予评价。

2.3.2 居民心理分析

①针对居民而言，对自己想要反映的问题，希望政府能够快的准确的给予自己答复，那么答复时间则是居民比较关心的问题。

②居民反映的问题，即居民想要解决的问题。居民反映的问题可能是自己身边的也可能是其他地方的，在一个留言中可能会反映不同地方的同一个问题和一个地方的不同问题。

③对于居民来说，反映后的问题是需要政府出面解决的，而希望在回复中给自己一个可靠的答案，而不是草草敷衍的内容。

2.3.3 回复内容评价指标

①留言时间和答复时间的差作为回复积极性指标。

②对于回复的内容，我们需要判断回复的内容中是否将居民提的所有问题都涉及到，将其作为答复完整性指标。

③每条答复内容中的地点与问题是否与留言相匹配，将其作为回复准确度指标。

④答复内容的可解释性指标。

2.3.4 各类指标计算方案

①答复积极性指标 h

h = 答复时间-留言时间

②居民在留言中通常会涉及到主要体现地点和问题关键性词语，我们提取出留言问题关键字储存到列表 list_1 中，提取答复内容关键字存储到 list_2 中，将 list_1 中的关键字逐个在 list_2 中循环匹配，将匹配到的个数存储在列表 list_3 中。

则答复完整性∂：

$$\partial = \frac{len(list\_3)}{len(list\_1)}$$

③为了检查该答复是否为答非所问，即答复的准确性，我们提取留言中的地址关键词，为对应的问题关键词，将地点和问题进行组合保存到 csv 文件中，对答复内容进行相同操作，计算两个文本相似度β；将其作为答复准确性指标。

④针对答复内容构建话题模型，通过话题模型，直接用话题词云来作为答复的解释信息，利用各类指标进行答复内容的评价。

参考文献

[1] 史会峰,卢艳霞.基于多项式分布模型的 Web 文本分类[J].华北电力大学学报,2003(06):83-85.
[2]贾利娟,刘娟,王健,周国民.基于 PyEcharts 的全球玉米贸易数据可视化系统建设及应用展望[J].农业展望,2019,15(03):46-54.
[3]武永亮,赵书良,李长镜,魏娜娣,王子晏.基于 TF-IDF 和余弦相似度的文本分类方法[J].中文信息学报,2017,31(05):138-145.
[4]姜阳,房龙.混淆矩阵算法在质检工作中的应用[J].经纬天地,2019(01):5-7.
[5]李海磊,杨文忠,李东昊,温杰彬,钱芸芸.基于特征融合的 K-means 微博话题发现模型[J].电子技术应用,2020,46(04):24-28+33.
[6]董小国,甘立国.基于句子重要度的特征项权重计算方法[J].计算机与数字工程.2006(08)
[7]任远航.面向大数据的 K-means 算法综述[J/OL].计算机应用研究:1-7[2020-05-04].
[8]可解释性推荐系统综述 Explainable Recommendation: A Survey and New Perspectives.2019-06-01.