智慧政务中的文本挖掘

摘要

随着互联网的飞速发展，文本数据库得以迅速增长，人们迫切需要有效的数据挖掘工具从海量文本数据中提取有价值的知识。党的十九届四中全会提出要创新行政管理和服务方式，加快推进全国一体化政务服务平台建设，这显然是对电子政务提出的进一步发展要求。互联网应用使得群众反映问题的渠道多样化，各类社情民意相关的文本数据不断增加，给人工进行留言划分和热点整理的部门带来了极大的挑战，所以建立智慧政务系统对政府办事能力的提升和效率的增加有极大的帮助。

对于本次赛题给出的智慧政务问题，我们使用了机器学习中的聚类分析，文本向量化等方法，同时也利用了 TF-IDF 以及 pipeline 管道机制等方法，从题中给的 excel 表格中提取需要数据，并深度挖掘了其中的隐藏数据，达成了题目要求的：群众留言分类，热点信息挖掘以及留言的回复评价这三个内容。同时，我们在热点信息挖掘和留言回复评价这两个问题中分别建立了模型，对于前者，我们建立了一个通过多个因素指标以及一定的权重加权平均从而判断热度的模型；后者我们建立了一个关于留言的评价系统，同样通过加权平均从而达到多个因素不同比重的效果。

最后我们也总结了模型中的不足之处，如留言评价体系中，情绪分析的应用可以帮助评价体系更加完整，我们对此进行了算法上面的修改，丰富了模型中的因子，同时也重新衡量了因子的权重，达到了更好的评价效果。

【关键词】TF-IDF；智慧政务；聚类分析；文本向量化；自然语言处理技术

Abstract

With the rapid development of the Internet, text databases have grown rapidly, and people urgently need effective data mining tools to extract valuable knowledge from massive text data. The Fourth Plenary Session of the Nineteenth Central Committee of the Party proposed to innovate administrative management and service methods and accelerate the construction of a national integrated government service platform. This is obviously a further development requirement for e-government. Internet applications have diversified the channels for people to reflect problems, and various types of social data and public opinion-related text data continue to increase, which poses great challenges to departments that manually divide messages and organize hotspots. The improvement and the increase of efficiency are of great help.

For the smart government issues given in this competition question, we used clustering analysis, text vectorization and other methods in machine learning. At the same time, we also used TF-IDF and pipeline

pipeline mechanism and other methods, from the excel table given in the question. It extracts the required data and deeply digs the hidden data in it, and meets the requirements of the topic: the classification of the mass message, the mining of hot information and the evaluation of the reply of the message. At the same time, we have established models in the two issues of hotspot information mining and message reply evaluation. For the former, we have established a model that judges the popularity by multiple factor indicators and a certain weight weighted average; the latter we have established An evaluation system on message, also through weighted average to achieve the effect of different factors with different proportions.

Finally, we also summarized the deficiencies in the model. For example, in the message evaluation system, the application of sentiment analysis can help the evaluation system be more complete. We have modified the algorithm above to enrich the factors in the model and also re-measure The weight of the factor is achieved, and a better evaluation effect is achieved.

Key words: TF-IDF; smart government affairs; cluster analysis; text vectorization; NLP

1 引言

随着互联网的飞速发展，文本数据库得以迅速增长，人们迫切需要有效的数据挖掘工具从海量文本数据中提取有价值的知识。党的十九届四中全会提出要创新行政管理和服务方式，加快推进全国一体化政务服务平台建设，这显然是对电子政务提出的进一步发展要求。互联网应用使得群众反映问题的渠道多样化，各类社情民意相关的文本数据不断增加，给人工进行留言划分和热点整理的部门带来了极大的挑战，所以建立智慧政务系统对政府办事能力的提升和效率的增加有极大的帮助。

使用计算机对各类社情民意的文本数据进行挖掘主要需要用到机器学习中的自然语言处理技术（NLP）。自然语言处理是计算机科学、人工智能和语言学的交叉领域，简单来说，就是开发能理解人类语言的应用程序或者服务。近年来，自然语言处理作为人工智能的一个重要领域得到了飞速发展，构建以自然语言处理为基础的智慧政务文本挖掘模型，解决社情民意文本数据的分类问题、热点整理问题以及回复评价问题，无疑可以解决相关部门的燃眉之急。

对于问题一，群众留言分类，需要建立关于留言内容的一级分类模型，即通过对留言内容进行提取和处理，之后与给出的内容标签体系进行相似度对比，选择相似度最高的内容对应的一级标签。之后计算该模型的精确率和召回率，通过F-Score对分类模型进行评价，根据评价结果进一步完善模型。

对于问题二，热点问题挖掘，即从大量文本数据中指导某一时段内群众集中反映的某一问题。这一问需要在提取对应表格中的文本信息后对其进行处理，并根据地点、人群进行归类，计算问题的持续时间、同类问题数量、点赞数和反对数等因素的权重并建立模型计算出各问题的热度指标进行排序并按题目要求输出表格。

对于问题三，对答复意见的评价。我们主要从答复的相关性、完整性、可解释性、答复时间以及情感五个方面进行分析，通过一定的标准对答复进行这些方面的评分，再根据这五个方面因素的权重建立模型给每个回复打分，从而达到对答复意见的评价这一目的。最后对该打分模型进行评价，进一步提升打分的准确性和公正性。

本文通过解决题目中的问题一、二、三，建立了一个使用NPL对社情民意文本数据处理的模型，通过建立该模型，达到了互联网政务的一部分要求。

2 准备工作

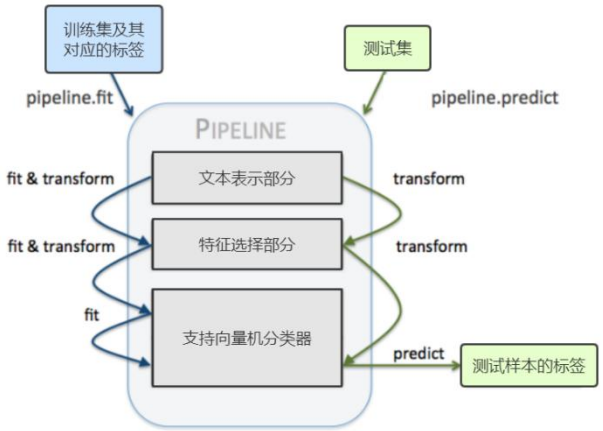
2.1 主要的库的准备

1. re 库：利用正则表达式去除中文数据中少量非文本数据
2. jieba 库：利用 jieba 库对文本进行分词处理
3. pandas 库：进行数据读取和写入 csv 文件，进行 dataframe 的数据遍历。
4. sklearn 库：对文本数据进行降维、分类和聚类等处理
5. pyhanlp 库：java 中的数据包，可在 python 中调用

2.2 相关知识准备

2.2.1 pipeline 管道机制

Python 中的 sklearn.pipeline.Pipeline()函数可以把多个学习器组成流水线，中间部分为转换器，最后一步为估计器。对于本题而言，我们利用该机制，将文本表示、特征选择部分和分类器组建在一起，当把训练集输入时，首先由文本表示部分在训练集上执行 fit 和 transform 方法，得到的数据将会传递给下一步，即特征选择部分，并且它同样执行 fit 和 transform 方法，最终转换后的数据会被传递给支持向量机分类器。流程如图所示。



2.2.2 macro_F1 值

该指标可用于评价多分类模型的效果，其计算公式为

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

其中 P_i 为第 i 类的查准率，R_i 为第 i 类的查全率。

2.2.3 TF-IDF：

TF-IDF 即词频-逆文本频率，其中 TF 为词频，表示某一字词在文本中出现的频率，IDF 为逆文本频率。TF-IDF 可用于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。字词的 TF-IDF 值越大，则该字词越重要。

2.2.4　卡方统计量：

卡方统计量（CHI）是常用的特征选择方法之一，能够衡量特征与类别之间的独立性。

法中使用最广泛的方法。特征词 w_r 的 χ^2 值为

$$\chi^2 = \sum_{i=1}^{2}\sum_{j=i}^{T}\frac{(Nn_{ij} - n_{i+}n_{+j})^2}{Nn_{i+}n_{+j}},$$

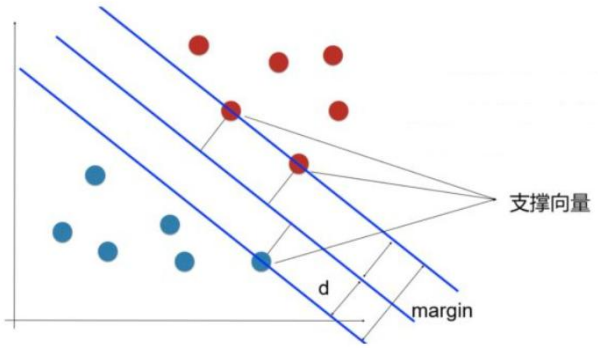
其中：T 为类别总数；n_{1j} 为在 c_j 类中包含特征词 w_r 的文本数；n_{2j} 为在 c_j 类中不包含 w_r 的文本数；n_{1+} 为数据集中包含 w_r 的文本数；n_{2+} 为数据集中不包含 w_r 的文本数；n_{+j} 为 c_j 类的文本数。若 χ^2 值较大，则说明特征 w_r 与类别 c_j 的相关性较大，反之亦然。

2.2.5　支持向量机

支持向量机是一种有监督的学习方法，在文本分类方面应用广泛。它是一种二分类模型，若要将其应用于多分类问题，常见的解决思路是将多分类问题分解为多个二分类问题进行求解，通过决策函数确定分类结果。

支持向量机的基本思想是找出一个超平面，可以使得各类样本点到该超平面的距离最远，即找出最大间隔超平面。

示意图如下所示。



超平面可由以下方程来表示：

$$w^T x + b = 0$$

而 margin(b, w)则表示超平面与样本点之间的最小距离，实际上，我们的优化目标就是要使这个距离最大化。

对于线性不可分的情况，可以通过引入核函数，将二维线性不可分样本映射到高维空间中，让样本点在高维空间线性可分。

2.2.6　HanLP

HanLP 是 java 里面的工具包，可以通过加载环境在 python 中调用，它支持中文分词（N-最短路分词、CRF 分词、索引分词、用户自定义词典、词性标注），命名实体识别（中国人名、音译人名、日本人名、地名、实体机构名识别），关键词提取，自动摘要，短语提取，拼音转换，简繁转换，文本推荐，依存句法分析（MaxEnt 依存句法分析、神经网络依存句法分析）。提供 Lucene 插件，兼容

Solr 和 ElasticSearch。

2.2.7 余弦相似度：

余弦相似度是通过计算两个向量的夹角余弦值来度量它们的相似性。余弦值越接近 1，则这两个向量越相似。假定有两个向量 A、B，余弦相似度的计算如下所示：

$$\text{similarity} = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}.$$

其中 A_i、B_i 分别为向量 A、B 的各分量。

3 群众留言分类

3.1 问题重述与分析

在问题一中，需要给群众的留言添加分类，需要利用附件 1.xlsx 中的三级标签和内容的对应，将附件 2.xlsx 中需要进行分类的文本数据进行分词和特征提取后，利用 sklearn 库对文本进行聚类处理，随后再根据聚类结果对返回文本内容，将三级标签对应的一级标签写入原附件 2.xlsx 中分类标签的那一类，即完成题目要求。

本题的研究路径设计分为四个部分：数据预处理、文本表示、特征选择以及构建分类模型。

3.2 数据预处理

3.2.1 分词

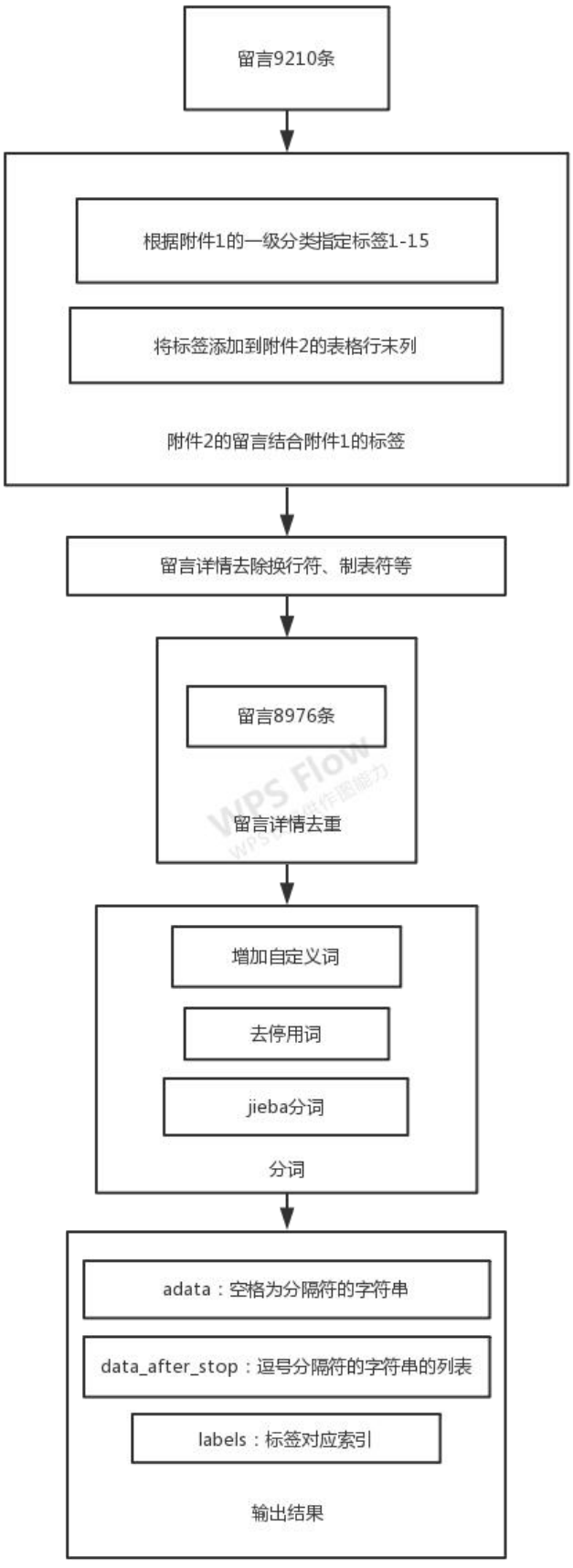
中文文本挖掘的基础和前提是对其进行分词处理。英文是以字母构成的单词为基础进行造句，计算机易于处理英文句子，可以将其分割为逐个单词进行挖掘处理。而相对的中文文本，是以汉字作为最小单位的，但是每个汉字不一定能构成特定的意思，需要由几个汉字组成词语进行语义划分。中文并不具备英文空格之类的词划分界限，只能通过一定的处理将没有划分标志的汉字字符串分割为符合句义、语义的词串，这是中文分词的要点也是难点所在。

本文采用了 python 中一个著名的中文分词模块——jieba 分词，对问题一中的留言详情进行划分处理。jieba 作为一个功能强大的分词库，录入了两万余条词语的基本库，其实现原理较为完善，设计的算法有基于前缀词典的有向无环图、动态规划、HMM 模型等。同时，jieba 库支持三种分词模式：精确模式、全模式、搜索引擎模式，本文采用的是精确模式。

3.2.2 去停用词

在进行完分词处理后，我们会遇到一些对实际表达无意义的词语，例如：“我、你、他、的、了”等。而在实际过程中，为了避免其对之后处理语段数据造成污染，我们会将其删去。我们通过读取一个常用的停用词词典，将分词后的词串中的停用词删除。

3.2.3 文本预处理流程图


留言9210条
根据附件1的一级分类指定标签1-15
将标签添加到附件2的表格行末列
附件2的留言结合附件1的标签
留言详情去除换行符、制表符等
留言8976条
留言详情去重
增加自定义词
去停用词
jieba分词
分词
adata：空格为分隔符的字符串
data_after_stop：逗号分隔符的字符串的列表
labels：标签对应索引
输出结果

3.2.4　数据预处理步骤

Step 1.转存文件并读取数据

由于题目提供数据是 excel 格式，python 对 excel 格式的数据的读取缓慢，所以需要先将附件 1.xlsx 和附件 2.xlsx 文件转存为对应的 csv 文件，即 1.csv 和 2.csv，提升 python 读取数据的速度。在 python 中提取 1.csv 内容为 data_1，提取 2.csv 内容为 data。

Step 2.对数据进行合并

对 data_1 中的一级分类标签进行提取，随后利用 drop_duplicates()对一级分类标签进行去重处理，保存在 labels 中。再利用动态命名方法，将 data 按照一级分类标签划分为 label_i（其中 i=1…15），并在对应行末列加上数字标签 i，最后将 label_i 进行拼接成 df。

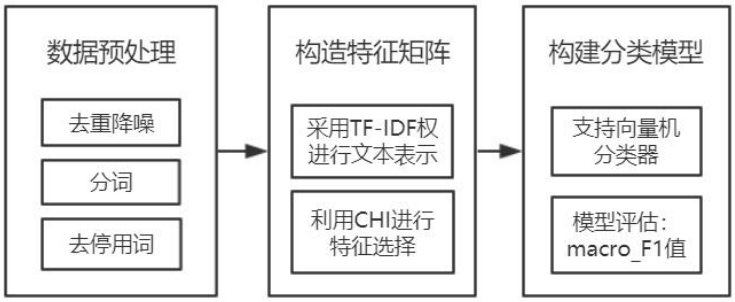
Step 3.对数据进行预处理

将 df 中的留言详情列提取出并命名为 content。对 content 的每一行利用 strip()去除换行符、制表符，再对其使用 drop_duplicates()进行去重处理。然后对留言详情进行 jieba 分词。最后对留言详情去除停用词，并将结果储存为 data_after_stop，同时储存一个空格分隔符的字符串格式的 adata、储存索引对应的数字标签 labels。

3.3 给群众留言分类并添加标签

原理：

预处理后的文本需要转换成计算机能够理解的形式，此处，我们利用 TF-IDF 权将其转化为向量。然后通过卡方统计量（CHI）进行特征选择，从而构造出最终的特征矩阵。最后，采用支持向量机分类器进行分类，将所得的特征矩阵作为分类模型的输入，训练该分类模型，并使用 macro_F1 值对该模型进行评估。



算法步骤：

Step1. 利用 pipeline 管道机制，将文本表示、特征选择部分和分类器组建在一起，构成一个模型

Step2. 将预处理后的文本作为训练数据，训练 step1 所得模型

Step3. 计算 macro_F1 值来进行评估

3.4 成果展示

3.4.1　分类成果

留言编号	留言用户	留言主题	留言时间	留言详情	一级标签
24	A00074011	建筑集团占道施工有劳	20/1/6 12:09:	围墙内。每天尤其	城乡建设
37	U0008473	大厦人为烂尾多年，劳	20/1/4 11:17:	着，不但占用人行道	城乡建设
83	A00063999	市A1区苑物业违规收伤	19/12/30 17:06	多次向物业和社区	城乡建设
303	U0007137	南路A2区华庭楼顶水箱	19/12/6 14:40:	，霉是一种强致癌	城乡建设
319	U0007137	2区华庭自来水好大一	19/12/5 11:17:	，霉是一种强致癌	城乡建设
379	A00016773	市盛世耀凯小区物业无	19/11/28 9:08:	业不是为业主服务	城乡建设
382	U0005806	询A市楼盘集中供暖一	19/11/27 17:14	月亮岛片区近年规划	城乡建设
445	A00019209	西路可可小城长期停水	19/11/19 22:39	帮助至今没有找到	城乡建设
476	U0003167	议取城市垃圾处理费不	19/11/15 11:44	在的物业公司也未经	城乡建设
530	U0008488	A3区魏家坡小区脏乱差	19/11/10 18:59	让人好好休息一下	城乡建设
532	U0008488	A市魏家坡小区脏乱差	19/11/10 12:30	让人好好休息一下	城乡建设
673	A00080647	四届非法业委会涉嫌侵	19/10/24 11:29	责令B4区有关部门	城乡建设
994	U0005196	梅溪湖壹号御湾业主用	19/9/18 22:43:	别的城市都已经一	城乡建设
1005	U0006509	翡翠湾强行对入住的业	19/9/18 13:36:	地产公司和金晖物	城乡建设
1110	A00099772	市锦楚国际星城小区三	19/9/9 11:07:	是无通知，突然断	城乡建设
1309	U0005083	和紫都用电的问题能不	19/8/21 15:12:	之后，我们的用电	城乡建设
1440	A0003288	际新城从6月份开始停	19/8/6 10:28:	的生活，而且我们	城乡建设
1775	U0002150	区区南西片区城铁站设	19/7/4 18:52:	A市，并且规划有	城乡建设
1783	U0004763	区府加大对滨水新城的	19/7/4 14:25:	或者几个半大小树	城乡建设
1827	U000613	区楚府线几个小区经常	19/7/1 20:14:	已停电三次。说是	城乡建设
2603	A00099650	用及西地省辉东安建工	19/4/20 16:50:	不出。去年8月，我	城乡建设
3607	A00046529	水嘉园1栋三单元群租	19/1/8 10:08:	急患，投诉给物业公	城乡建设
3742	A00013884	小区外的非法汽车检测	18/12/26 10:13	备检定（省级）；方	城乡建设

Sheet1

注：摘自附件中分类成果.xlsx 中

3.4.2　模型评价结果

```
Out[7]:  0.910030097248191
```

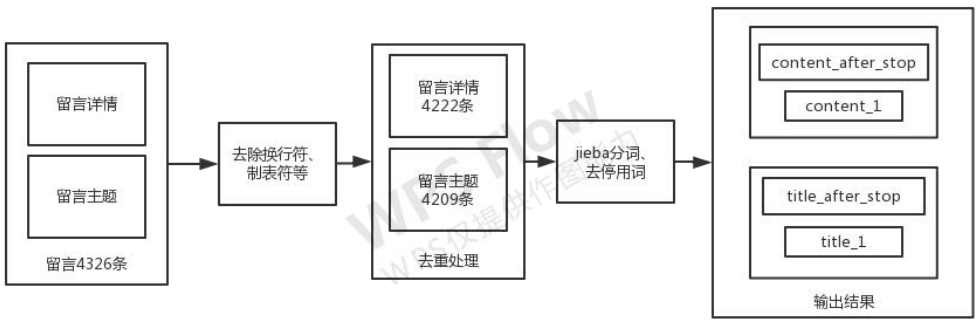
4　热点问题挖掘

4.1 问题重述及分析

在问题二中需要先对整个表格的数据进行预处理，然后将数据进行聚类。根据每一类问题的数量、问题的点赞数和反对数，以及问题的时间根据一些文献的内容衡量建立出每个指标的权重，并建立模型根据这个权重计算出每一类问题热度并排行。随后根据前五的问题返回到原表格中的留言，并将留言写入到热点问题留言明细表中。之后建立热点问题表，写入热度指数、时间等参数，并通过命名实体识别提取化简地点和人群，通过 python 将文本摘要并写入热点问题表中。

4.2 数据预处理

4.2.1　文本预处理流程图



4.2.2 数据预处理步骤

Step 1.转存文件并读取数据

由于题目提供数据是 excel 格式，python 对 excel 格式的数据的读取缓慢，所以需要先将附件 3.xlsx 文件转存为对应的 csv 文件，即 3.csv，提升 python 读取数据的速度。在 python 中提取 3.csv 内容为 data。

Step 2.对数据进行预处理

将 data 中的留言详情列、留言主题列分别提取出并命名为 content、title。对 content、title 的每一行利用 strip()去除换行符、制表符，再对其使用 drop_duplicates()进行去重处理。然后联系附件中的留言详情、留言主题，我们自定义了一个分词字典，将其载入后对留言详情、留言主题进行 jieba 分词。最后去除停用词，并将结果储存为 content_after_stop、title_after_stop，同时储存空格分隔符的字符串格式的 content_1、title_1。

4.3 聚类

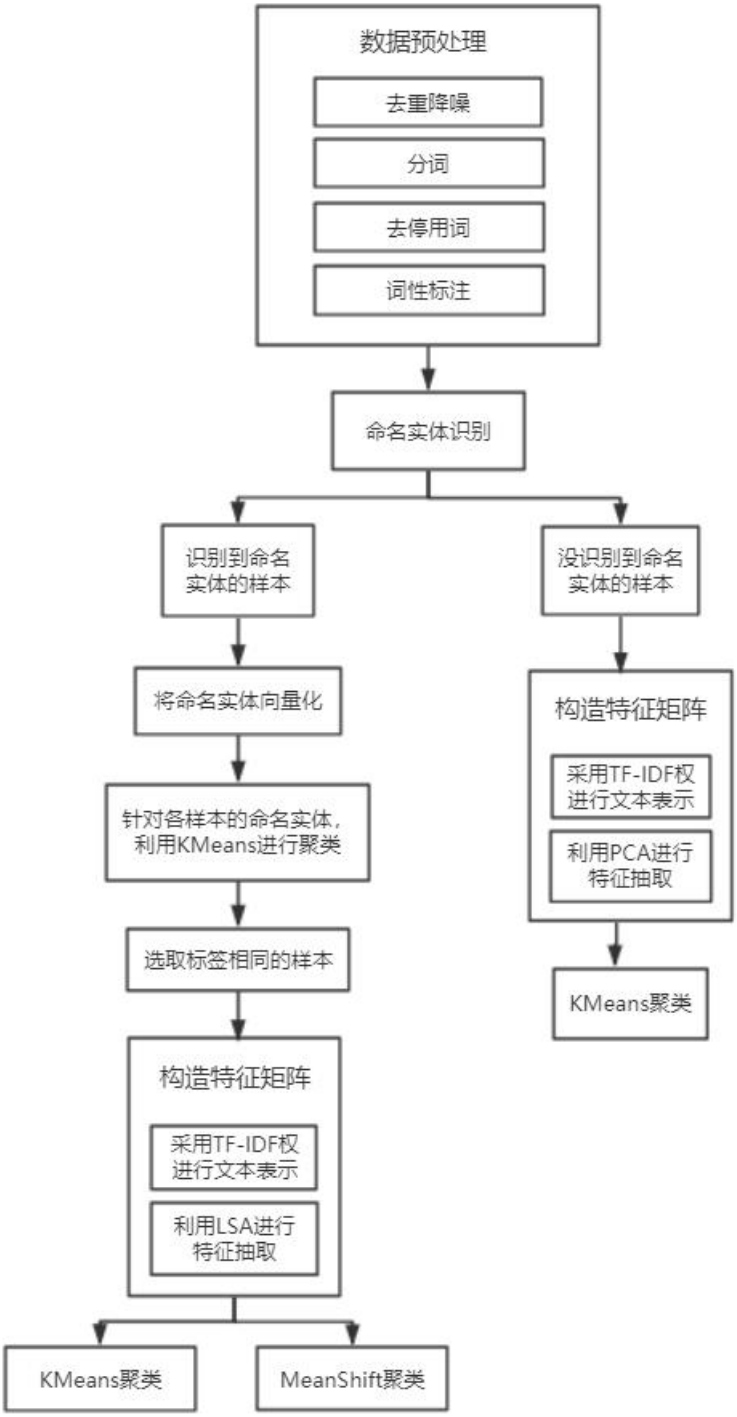
首先我们需要把描述同一个问题的留言归为一类,考虑到群众在反映问题时,都会指出其发生的地点等，那么，关于同一问题的大部分留言的标题及详情就会有相似的命名实体，故先对各样本的命名实体进行聚类，将含有相似的命名实体的样本归为一类,然后对每一类别的样本再次进行聚类。由于留言详情比较冗杂，此处选取留言标题进行处理。

获取分词、去停用词后的留言标题,利用 pyltp 给标题中的每个词标注词性，然后继续使用 pyltp，对各标题进行命名实体识别，从而提取出标题中的命名实体。但存在某些标题没有识别出命名实体的情况，因此需将样本集合分成两个：其中一个样本集合中的样本都是有识别到命名实体的，另一个则没有。

对于没有识别到命名实体的样本，采用 TF-IDF 权进行文本表示，利用主成分分析（PCA）进行特征抽取，从而构造出特征矩阵作为聚类模型的输入，这里使用 KMeans 算法进行聚类。

对于识别到命名实体的样本，先通过 TF-IDF 权将其命名实体转化为向量，将所得特征矩阵传入 KMeans 聚类模型进行聚类。获取聚类后各样本的标签，然后选取标签相同的样本，对每类样本利用 TF-IDF 权进行文本表示，若特征维度较大，则使用隐含语义索引（LSA）进行特征抽取，最后采用 KMeans 算法以及 MeanShift 算法对每一类样本再次进行聚类，其中主要使用 MeanShift 算法，而 KMeans 算法用于样本数量较多的类别。

具体流程如下图：



算法步骤：

Step1. 获取分词、去停用词后的留言标题，进行词性标注

Step2. 对标题进行命名实体识别

Step3. 提取各标题的命名实体

Step4. 选取有识别到命名实体的标题

Step5. 将 step4 所选标题的命名实体向量化

Step6. 将 step5 所得矩阵传入 KMeans 聚类模型进行聚类

Step7. 获取聚类后各样本的标签

Step8. 选取标签为 i 的标题，其中 i=0，1，2，...，146

Step9. 采用 TF-IDF 权对标题进行文本表示

Step10. 判断特征数量是否大于 100，若大于 100，则使用 LSA 进行特征抽取

Step11. 将所得的特征矩阵输入聚类模型进行聚类，主要用 MeanShift 算法，特别地，对于一些样本数量较多的类别，则使用 KMeans 算法

Step12. 重复 step8 到 step11，直到 i 达到 146

Step13. 选取没有识别到命名实体的标题

Step14. 采用 TF-IDF 权对标题进行文本表示

Step15. 利用 PCA 进行特征抽取

Step16. 将所得的特征矩阵输入 KMeans 聚类模型进行聚类

4.4 热度评价模型

4.4.1 热度评价总述

热点问题的定义为某一时段内群众集中反映的某一问题，因此我们希望确定哪些话题是在一段时间内受到大量关注与支持，将这些话题作为网络问政的舆情热点。我们通过参考一些文献以及结合实际情况，将热度评价指标划分为以下几个要素进行量化计算：数量占比（考虑到在此时段的占比以及总时段的占比）、留言速度、点赞占比、留言人数占比。通过量化考量，我们可以将较为抽象的“热点问题”转化为较详细具象的“热度指数”，便于政府进行舆情考察时更深入了解哪些问题更加聚焦于公民身上。

4.4.2 数量占比

数量占比是能够最直接体现热度的一个指标。一个话题若想成为热点，首先需要有广泛的数量。我们将数量占比划分为此时段占比、总时段占比。此时段占比指的是在该话题期间内，该话题的留言数占该时间段内总留言数的比例，这样能更好地体现该时段该话题的重要程度以及集中度(指该话题是否在该时段内受到大量关注)。总时段占比指的是该话题的留言数与总留言数之比，这个比例衡量的是在整个数据集中该话题的重要程度。两者可以很清晰看到差别，前者更侧重于群众集中反映的问题，后者更侧重于问题的总体量。

$$Weight_{time} = \frac{n}{N}$$

$$Weight = \frac{n}{M}$$

$Weight_{time}$是此时段占比，$Weight$ 是总时段占比。其中，n 代表该话题的留言总数，N 代表该话题时间段内的留言数，M 代表整个数据中的留言总数。

4.4.3 留言速度

由于热点问题指的是某一时段内集中反映的问题，因此一个热点问题应当具有较高的留言速度。若一个话题相关的留言数目总体很多，但其却分散在各个时间段内，平均到每个单位时间的数目较少，这说明这个问题虽还未解决但并不能作为网络问政中的热点问题。例如一个在 7 天内有 50 条留言的话题热度要远远高于一个在 2 年内有 50 条留言的话题热度。因此留言速度是我们需要考虑的指标。

$$Speed = \frac{n}{\Delta t}$$

$Speed$ 是留言速度，其中，n 代表该话题的留言总数，Δt 代表该话题的时间间隔。我们将 3 天记为时间间隔的单位时间。

4.4.4 点赞占比

很多时候作为社会公民的我们，遇到想要上访询问的问题，会先在网站上浏览是否有相关的问题，若有就点个赞，若没有再去考虑自己留言询问。因此点赞数是我们需要考虑的一个重要指标，它反映了群众对某些问题的关注程度与喜好程度。并且我们没有单独考虑点赞数，同时也将反对数作为热度计算的一个考量，而不是单纯的相加减，因为反对有可能是由一些针对此问题的人进行的评价。比如有较多群众反映的学院变相强制实习的问题、小区附近夜宵摊空气污染的问题，都拥有较多的反对数。我们可以适当猜想，这些是否为学院的相关人士、夜宵摊的店主等所致？有可能是他们看到群众投诉自己，因此点了反对。所以我们没有机械的将点赞数减去反对数进行测度，而是采用如下公式：

$$Like\ weight = \frac{\alpha \times Like + \beta \times Oppose}{\alpha \times Like_{sum} + \beta \times Oppose_{sum}}$$

$Like\ weight$ 是点赞占比，其中，$Like$ 代表该话题的点赞数，$Oppose$ 代表该话题的反对数，$Like_{sum}$代表该话题时段内的点赞总数，$Oppose_{sum}$代表该话题时段内的反对总数。而α和β是调节因子，实验中我们取$\alpha = 0.7, \beta = 0.3$ 进行计算。

4.4.5 留言人数占比

热点问题往往是大多数人共同反映的问题，即使一个问题由 1 个人反映了 5 次，也不如 5 个人分别反映 1 次更具有热度。因此我们可以通过留言用户的 ID 计算留言人数占比。

$$People\ weight = \frac{p}{P}$$

$People\ weight$ 是留言人数占比，其中，p 代表该话题留言人数，P 代表总留言人数。

4.4.6 模型计算

通过对以上各个要素的归一化处理和加权求和，我们可以得出以下热度指数计算公式：

$$Hot\ degree = 0.5 \times Weight + 0.2 \times Like\ weight + 0.1 \times Weight_{time} + 0.1 \times Speed + 0.1 \times People\ weight$$

4.4.7 算法步骤

Step1.数据处理

首先读取聚类结果的文件，并将其中的“留言时间”列的时间格式统一为 datetime，方便后来的处理。提取类别标签并计算总留言数 M、总留言人数 P（通过对“留言用户”列进行去重计算）。

Step2.计算每类话题的具体分值

对标签列表中每个标签进行遍历。如设标签为 a，首先提取标签为 a 的数据，计算标签 a 的留言数 n、留言人数 p、时间间隔Δt（单位时间为 3 天）。若时间间隔小于 3 天，我们将其热度指数定为 0。然后统计标签 a 的点赞数 $Like$、反对数 $Oppose$，以及标签 a 留言时间段内的点赞总数$Like_{sum}$、反对总数$Oppose_{sum}$、留言总数 N。为了防止点赞占比分母为 0 无意义，我们定义点赞总数、反对总数之和为 0 的点赞占比为 0。最后计算出标签 a 的总时段人数占比 $Weight$、点赞

占比 $Like\,weight$、此时段人数占比$Weight_{time}$、留言速度 $Speed$ 以及留言人数占比 $People\,weight$。并按标签的顺序添加到各分值列表。

Step3.热度指数计算与排序

将 5 个具体分值列表进行归一化处理，并加权求和得到热度指数数组。生成一个表格，列名为“labels”、“hot_degree”，将每个标签与热度指数相对应，并通过排序得到热度指数最高的前五个标签。

Step4.生成具体 Excel 表格

利用 for 循环将排序后的五个标签进行遍历，将问题 ID、具体的留言信息按格式添加到表格中，这里注意到要对问题 ID 先排序，再对留言编号进行排序。最后生成热点问题留言明细表.xls。

对于热点问题表，我们在此模块仅生成一个表格样式，以及具体的热度排名、问题 ID、热度指数、时间范围，地点人群以及问题描述的提取由下一模块完成。注意到这里的时间范围的提取要对时间数据格式化处理。

4.5 生成热点问题表

4.5.1 命名实体识别提取人群和地点

库的准备：

Python 中命名实体识别的库主要有 nltk 库和 pyhanlp 库，本文中采取了 pyhanlp 进行命名实体识别。HanLP 是一系列模型与算法组成的 java 工具包，功能强大，在 python 中调用 HanLP 需要先进行库的安装和环境的安装。

完成安装后第一次运行 pyhanlp 需下载.jar 和 data 文件，这个可以手动下载之后放到下载的位置，再运行 pyhanlp 进行解压和读取即可。

原理：

由于中文文本中词语词之间没有分隔符，所以中文文本的分词和命名实体识别密不可分。由于中文文本的命名实体互相嵌套，所以需要在一个集成的框架下进行命名实体识别，并达到整体的最优效果。

利用层叠隐马尔可夫模型，将人名、地名、机构名等命名实体识别融合到一个相对统一的理论模型首先需要利用底层尹默尔科夫模型识别出普通无嵌套的人名、地名和机构名等，在采取高层隐马尔可夫模型识别出嵌套的人名、地名、机构名。

将人名、地名、机构名分层标记，之后单独取出储存到元组，最后在将几个元组合并，输出统一的命名实体表。利用元组的提取功能，提取出地点和人群因素，将元组中的因子去重，化简，输出到表格指定位置

算法步骤：

Step1.获得系统的 jvm 路径，并打开 jvm 虚拟器，设定虚拟器运行空间。

Step2.定义函数进行先对文本进行繁体和简体的转换，这一步是为了下面的统一处理

Step3.定义函数先将句子进行切词和粗略的词性标注，其实也就是粗略的命名实体识别

Step4.分别定义函数识别地名、人名、机构名，并对其进行特殊的词性标注

Step5.定义函数将上述单独的识别结果转化成列表，便于提取。并定义函数写出时间实体

Step6.定义函数返回单一实体类别的列表，并将单一实体结果汇总成列表方便提

取。
Step7.利用循环读取热点问题留言明细表.xls 中热度排行前五的具体留言内容，调用函数识别地名和人名。
Step8.提取地点和人名并返回对应的列表，对列表中元素进行去重和化简处理，并输出到 excel 表格指定的人群地名栏

4.5.2　提取留言内容摘要并将其写入 excel 中
原理：
利用 HanLP 的摘要功能将热度指数前五的留言分类提取摘要。
自动摘要算法最常见和最易实现的是 TF-IDF，但是 TextRank 的算法更好。TextRank 是在 Google 的 PageRank 的算法的启发下，针对文本里的句子设计的权重算法，自动摘要。它主要是利用投票的原理，让每一个单词给它的窗口投赞成票，票的权重取决于自己票数。同时又采用了矩阵迭代收敛的方式解决了这个悖论，将自动摘要变得合理。
TextRank 公式在 PageRank 公式基础上引入了边的权值的概念，可以代表两个句子的相似度。TextRank 公式如下

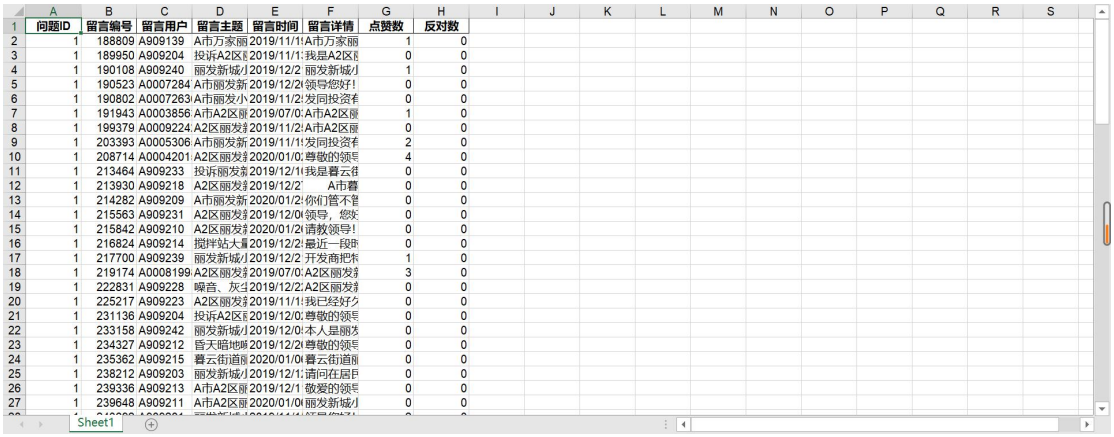
$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

算法步骤：
Step1.利用循环读取热点问题留言明细表.xls 中热度排行前五的具体留言内容，并将其格式转化成逗号分隔的字符串
Step2.对上述字符串进行断句处理，进行分词，并过滤停用词
Step3.计算 BM25 相关性矩阵
Step4.进行迭代投票
Step5.排序并输出结果，转换结果格式
Step6.将数据对应写入留言摘要并保存文件

4.6 成果展示
4.6.1　热点问题表

	A	B	C	D	E	F	G	H	I	J	K
1	热度排名	问题ID	热度指数	时间范围	地点/人群	问题描述					
2	1	1	0.633654	2019/02/0	A市A2区新城	A市暮云街道丽发新城社区搅拌站灰尘。					
3	2	2	0.308957	2019/07/0	A市新城	A市伊景园滨河苑捆绑销售车位。					
4	3	3	0.30579	2019/01/1	外滩A4区	A4区绿地海外滩二期业主被噪音扰得快烦死了。					
5	4	4	0.276636	2019/01/1	A3区	A3区郝家坪小学何时扩建。					
6	5	5	0.265002	2019/01/0	A7县	A7县东六路下穿长永高速在月底能否如期通车。					
7											
8											
9											
10											

注：摘自附录中热点问题表.xls
4.6.2　热点问题留言明细表



问题ID	留言编号	留言用户	留言主题	留言时间	留言详情	点赞数	反对数
1	188809	A909139	A市万家丽	2019/11/1	A市万家丽	1	0
1	189950	A909204	投诉A2区	2019/11/1	我是A2区	0	0
1	190108	A909240	丽发新城小	2019/12/2	丽发新城小	1	0
1	190523	A0007284	A市丽发新	2019/12/2	领导您好！	0	0
1	190802	A0007263	A市丽发小	2019/11/2	发同投资者	0	0
1	191943	A0003856	A市A2区丽	2019/07/0	A市A2区丽	1	0
1	199379	A0009224	A2区丽发新	2019/11/2	A市A2区丽	0	0
1	203393	A0005306	A市丽发新	2019/11/1	发同投资者	2	0
1	208714	A0004201	A2区丽发新	2020/01/0	尊敬的领导	4	0
1	213464	A909233	投诉丽发新	2019/12/1	我是暮云街	0	0
1	213930	A909218	A2区丽发新	2019/12/2	A市暮	0	0
1	214282	A909209	A市丽发新	2020/01/2	你们管不管	0	0
1	215563	A909231	A2区丽发新	2019/12/0	领导，您好	0	0
1	215842	A909210	A2区丽发新	2020/01/2	请教领导！	0	0
1	216824	A909214	搅拌站大量	2019/12/2	最近一段时	0	0
1	217700	A909239	丽发新城小	2019/12/2	开发商把车	1	0
1	219174	A0008199	A2区丽发新	2019/07/0	A2区丽发新	3	0
1	222831	A909228	噪音、灰尘	2019/12/2	A2区丽发新	0	0
1	225217	A909223	A2区丽发新	2019/11/1	我已经好久	0	0
1	231136	A909204	投诉A2区	2019/12/0	尊敬的领导	0	0
1	233158	A909242	丽发新城小	2019/12/0	本人是丽发	0	0
1	234327	A909212	昏天暗地邮	2019/12/2	尊敬的领导	0	0
1	235362	A909215	暮云街道丽	2020/01/0	暮云街道丽	0	0
1	238212	A909203	丽发新城小	2019/12/1	请问在居民	0	0
1	239336	A909213	A市A2区丽	2019/12/1	敬爱的领导	0	0
1	239648	A909211	A市A2区丽	2020/01/0	丽发新城小	0	0

Sheet1

注：摘自附件中热点问题留言明细表.xls

5　答复意见评价

5.1 问题重述和分析

问题三需要根据答复的内容建立一个模型，从几个方面进行对答复的评价。经过讨论，我们认为大致的评价标准如下：

答复的相关性可以从问题和对应的答复内容的重合程度进行判断；答复的完整性主要是判断答复是否依照一定的格式；答复的可解释性主要是答复是否发挥了一定作用；答复时间通过留言和答复的时间差判断；答复的情绪分析主要是利用情感词进行评判和打分；答复的详细程度主要从答复的长度判断。

评价标准选择好之后需要详细的对评价打分，随后根据每个评价标准的权重建立评价模型，对所有的答复意见进行评价。

5.2 评价标准

5.2.1　因素的标准

①相关性：利用留言详情与答复意见之间的语义相似度来衡量，相似度越高，则认为答复的内容和群众反映的问题越相关。

②完整性：完整性的考量可以判断政府是否按照一定的格式进行答复，而这可以评价政府的严谨程度、程序化程度，因此也是个重要的评价指标。由于并未查询到官方的答复格式文件，我们通过比较大量的留言答复意见，将完整性的考量划分为三个方面：开头、结尾、日期。

开头	1 表示有开头语；0 表示无开头语
结尾	1 表示有结束语；0 表示无结束语
日期	1 表示结尾有注释日期；0 表示结尾无注释日期
完整性 = 0.4*开头 + 0.4*结尾 + 0.2*日期	

③可解释性：可解释性衡量政府对该留言是否做出了反应，是否发挥了作用。我们将其划分为四个方面进行考量：调查过程说明、处理过程说明、处理依据说明、处理结果说明。

调查过程说明	1 表示有详细的调查过程说明；0 表示无详细的调查过程说明

处理过程说明	1 表示有详细的处理过程说明；0 表示无详细的处理过程说明
处理依据说明	1 表示有依据法律法规进行处理；0 表示没有依据法律法规进行处理
处理结果说明	1 表示有提供相关电话信息给群众进行回访；0 表示没有提供相关电话信息给群众进行回访；
可解释性 = 0.3*调查过程说明 + 0.3*处理过程说明 + 0.2*处理依据说明 + 0.2*处理结果说明	

④详细程度：利用答复意见的长度来衡量
⑥时效性：时效性测度了政府对群众留言的反应及处理是否及时，可以在很大程度上看出政府对民情的了解程度，以及群众的满意程度，从而评判答复的质量。我们通过计算留言时间、答复时间的时间差来对其进行测度。

5.2.2 权重的标准

由于相关文献比较少，所以我们采用了主观赋权法和客观赋权法相结合的方法，根据一部分的文献中的信息以及一部分的生活常识进行权重的确定，最终权重确定如下：

得分 = 可解释性（30%）+时效性（20%）+相关性（20%）+完整性（15%）+详细程度（15%）

这个权重的确定肯定了可解释性在整体评判中的重要性，这点在日常生活中我们也深有感触，如果对话双方完全不理解对方的意思，那么这无疑是最触怒建议人的一点。其次时效性和相关性相对来说也比较重要，是判断一个机构甚至政府在工作中的效率的重要因素，所以相对完整性和详细程度来说也会稍微权重高一点。政府工作中可以比较要求完整性，但是完整性即使不高，对收到回复的人来说，只要回复的及时且准确，也没必要太过苛刻，详细程度也是如此，其实作为收到回复的人来说，我们只是想知道结果，对整体的条例、法律等的认识的获取并没有那么大的期待，所以详细程度来说也会权重低一些。

5.3 模型建立

5.3.1 完整性：

原理：

我们利用 dataframe['列名'].str.contains('包含文字')函数对答复意见进行筛选。其中含有“您好”、“你好”、“如下”的答复判断为有开头语，含有“感谢”、“谢谢”的答复判断为有结束语。再利用 dataframe['列名'].str.endswiths('包含文字')对结尾含有“日”的答复进行筛选，判断为含有日期注释。最后再进行加权求和。

算法步骤：

Step1.读取附件 4.xlsx 并转换为 4.csv，通过 pd.read_csv 将 csv 转换成表格形式进行操作。

Step2.添加 3 个“0”列到表格中，并命名为“开头”、“结尾”、“日期”。

Step3.将答复意见含有“您好”、“你好”、“如下”的答复“开头”列赋值 1。

Step4.将答复意见含有“感谢”、“谢谢”的答复“结尾”列赋值 1。

Step5.将答复意见结尾含有“日”的答复“日期”列赋值 1。
Step6.将这 3 列进行加权求和，并储存为 completeness_array 数组中方便后面调用。

5.3.2 详细程度
原理：
利用答复意见的长度来衡量，一般而言，答复越长，说明、解释就会更详细。首先我们对答复进行预处理，去除字符、标点符号等，利用 jieba 对答复进行分词，注意到这里不能删去停用词，故不做去停用词处理，最后计算各答复的中文字符的个数，从而得到每个答复意见的长度。

算法步骤
Step1. 数据预处理：去除字符、标点符号，分词
Step2. 通过判断答复中各个元素的编码是否在常用中文字符的编码范围内，若在范围内，则计数加一，遍历整个答复意见，从而得到其长度

5.3.3 相关性
原理：
利用留言详情与答复意见之间的语义相似度来衡量，相似度越高，则认为答复的内容和群众反映的问题越相关。先选取预处理后的留言详情及其对应的答复意见，利用 TF-IDF 权将它们都转化为向量，若特征数量大于 10，则使用 LSA 进行特征抽取，得到文本的关键，并且降低特征的维度，从而可以提高后面计算相似度的效率。得到留言详情、答复的特征向量后，通过计算两个向量的余弦相似度来衡量相关性。

算法步骤：
Step1. 选取一条预处理后的留言详情及其对应的答复意见
Step2. 采用 TF-IDF 权对 step1 所得数据进行文本表示
Step3. 判断特征数量，若数量大于 10，则使用 LSA 对 step2 所得的特征向量进行特征抽取
Step4. 计算最终得到的两个特征向量的余弦相似度
Step5. 遍历整个样本集合，重复 step1 到 step4

5.3.4 可解释性：
原理：
我们利用 dataframe['列名'].str.contains('包含文字')函数对答复意见进行筛选。其中含有“核实”、“调查”、“考察”、“查”的答复判断为有详细的调查过程说明，不含有“转交”或者含有“转交”但同时含有“处理”、“目前”、“据”、“经”的答复判断为有详细的处理过程说明，含有“《”、“条例”、“依据”、“规定”的答复判断为有依据法律法规进行处理，含有“电话”、“致电”、“热线”、“拨打”的答复判断为有提供相关电话信息给群众进行回访。最后再进行加权求和。

算法步骤：

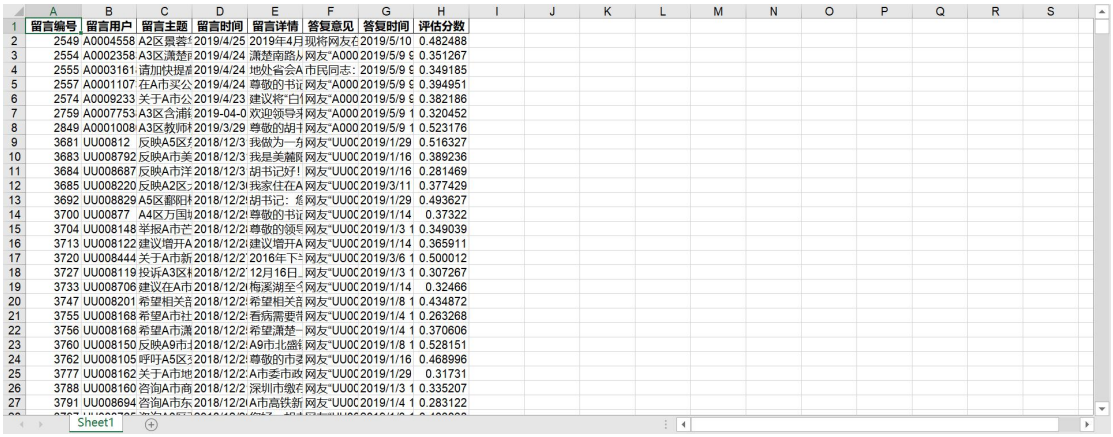
Step1.读取附件 4.xlsx 并转换为 4.csv，通过 pd.read_csv 将 csv 转换成表格形式进行操作。
Step2.添加 3 个“0”列到表格中，并命名为“调查过程”、“处理依据”、“处理结果”；再添加 1 个“1”列到表格中，并命名为“处理过程”。
Step3.将答复意见含有“核实”、“调查”、“考察”、“查”的答复“调查过程”列赋值 1。
Step4.将答复意见含有“转交”的答复“处理过程”列赋值 0。再将其中含有“处理”、“目前”、“据”、“经”的答复“处理过程”列赋值 1。
Step5.将答复意见结尾含有“《”、“条例”、“依据”、“规定”的答复“处理依据”列赋值 1。
Step6.将答复意见结尾含有“电话”、“致电”、“热线”、“拨打”的答复“处理结果”列赋值 1。
Step7.将这 4 列进行加权求和，并储存为 interpretability_array 数组中方便后面调用

5.3.5 时效性：
原理：
我们通过计算答复时间与留言时间的时间差来量化答复的时效性。
算法步骤：
Step1.读取附件 4.xlsx 并转换为 4.csv，通过 pd.read_csv 将 csv 转换成表格形式进行操作。
Step2.将留言时间、答复时间列转换为 datatime 格式的时间数据。
Step3.计算时间差，并储存到 diff_time 列中，并利用通过计算倒数来计算时效性。
Step4.将时效性列进行归一化处理，并储存到 timeliness_array 数组中。

5.3.6 总模型
原理：
通过查阅资料以及阅读具体的答复意见，我们将以上五个属性进行如下加权求和处理：
得分=可解释性（30%）+时效性（20%）+相关性（20%）+完整性（15%）+详细程度（15%）

算法步骤：
Step1. 导入各属性的模块。
Step2. 统一各属性的格式为 2816*1 的数组，并且进行归一化处理。
Step3. 将以上 5 个数组进行加权求和。
Step4.读取附件 4.xlsx 并转换为 4.csv，通过 pd.read_csv 将 csv 转换成表格形式进行操作。
Step5.将加权求和的得分写入表格最后一列“评估分数”，并生成相应的 xls 文件。

5.4 成果展示



留言编号	留言用户	留言主题	留言时间	留言详情	答复意见	答复时间	评估分数
2549	A0004558	A2区景荟	2019/4/25	2019年4月	现将网友在	2019/5/10	0.482488
2554	A0002358	A3区潇楚	2019/4/24	潇楚南路从	网友"A000	2019/5/9 9	0.351267
2555	A0003161	请加快提	2019/4/24	地处省会A	市民同志：	2019/5/9 9	0.349185
2557	A0001107	在A市买公	2019/4/24	尊敬的书记	网友"A000	2019/5/9 9	0.394951
2574	A0009233	关于A市公	2019/4/23	建议将"白	网友"A000	2019/5/9 9	0.382186
2759	A0007753	A3区含浦	2019-04-0	欢迎领导到	网友"A000	2019/5/9 1	0.320452
2849	A0001008	A3区教师	2019/3/29	尊敬的胡书	网友"A000	2019/5/9 1	0.523176
3681	UU00812	反映A5区	2018/12/3	我做为一名	网友"UU0C	2019/1/29	0.516327
3683	UU008792	反映A市美	2018/12/3	我是美麓居	网友"UU0C	2019/1/16	0.389236
3684	UU008687	反映A市洋	2018/12/3	胡书记好！	网友"UU0C	2019/1/16	0.281469
3685	UU008220	反映A2区	2018/12/3	我家住在A	网友"UU0C	2019/3/11	0.377429
3692	UU008829	A5区鄱阳	2018/12/2	胡书记：您	网友"UU0C	2019/1/29	0.493627
3700	UU00877	A4区万国	2018/12/2	尊敬的书记	网友"UU0C	2019/1/14	0.37322
3704	UU008148	举报A市芒	2018/12/2	尊敬的领导	网友"UU0C	2019/1/3 1	0.349039
3713	UU008122	建议增开A	2018/12/2	建议增开A	网友"UU0C	2019/1/14	0.365911
3720	UU008444	关于A市新	2018/12/2	2016年下	网友"UU0C	2019/3/6 1	0.500012
3727	UU008119	投诉A3区	2018/12/2	12月16日	网友"UU0C	2019/1/3 1	0.307267
3733	UU008706	建议在A市	2018/12/2	梅溪湖至今	网友"UU0C	2019/1/14	0.32466
3747	UU008201	希望相关	2018/12/2	希望相关	网友"UU0C	2019/1/8 1	0.434872
3755	UU008168	希望A市社	2018/12/2	看病需要	网友"UU0C	2019/1/4 1	0.263268
3756	UU008168	希望A市潇	2018/12/2	希望潇楚	网友"UU0C	2019/1/4 1	0.370606
3760	UU008150	反映A9市	2018/12/2	A9市北盛	网友"UU0C	2019/1/8 1	0.528151
3762	UU008105	呼吁A5区	2018/12/2	尊敬的市	网友"UU0C	2019/1/16	0.468996
3777	UU008162	关于A市地	2018/12/2	A市委市政	网友"UU0C	2019/1/29	0.31731
3788	UU008160	咨询A市商	2018/12/2	深圳市缴	网友"UU0C	2019/1/3 1	0.335207
3791	UU008694	咨询A市东	2018/12/2	A市高铁新	网友"UU0C	2019/1/4 1	0.283122

Sheet1

注：摘自附件中评估分数表.xls

5.5 模型优化

由于在上述模型中我们并未考虑情感方面的因素在评价中的作用，所以在评价模型时和优化模型中考虑了情感的因素对答复的评价作用，所以模型优化主要通过添加情绪分析以及群众再访率进行模型的完善。

5.5.1 情绪分析

原理：

进行情绪分析最简单的方法就是在语句中找到情感词，并判断情感词属于积极还是消极，因此我们先建立了两个情感词词典，一个是积极的，一个是消极的。但是由于情感的表述是分等级的，显然“很好”和“好”的情感等级不同，所以我们还考虑了情感等级，建立了程度级别词典。由于中文文本中存在否定词，且否定词会改变情感词属性（将积极变为消极），所以我们加入了否定词词典，用来更精确的判断情感词。除了对情感词、程度词、否定词的判断外，由于中文的语言习惯，感叹号表示比较强烈的情感，作用形同程度词，所以也要考虑感叹号的作用。

根据上述的判断方法，可以判断出一个分句的积极分值和消极分值，整句由许多分句构成，所以整句的积极分值和消极分值也可以进行计算。但是如果想判断一个句子是积极还是消极，还需要将积极分值和消极分值进行处理。文本中进行的处理是将积极分值和消极分值直接进行抵消，然后根据抵消后的数值判断句子的情感。数值为正，则标记为积极，并输出“1”；数值为负，则标记为消极，并输出“-1”；如果数值恰好抵消，则标记为中性，并输出“0”。

这些输出主要是为了方便之后对权重的计算。

算法设计：

为了方便调用，将算法定义成了函数。

Step1. 首先建立相关词典（情感词词典，否定词词典，程度词词典），并为程度词设定权值。

Step2. 查找分句的情感词，记录积极和消极，并记录情感词的位置。

Step3. 根据 Step2. 中的情感词位置，在情感词前查找程度词，找到就停止查找。根据之前定义的 Step1. 程度词的权值，将其乘以情感词对应的情感值。

Step4. 根据 Step2. 中的情感词位置，在情感词前查找否定词，找到所有的否定

词，如果数量是奇数，就将上一步骤里的情感值乘上-1，如果数量是偶数，则将上一步骤里的情感值乘上 1。
Step5.查找分句结尾是否有感叹号，如果有感叹号则往前寻找对应的情感词，如果有情感词就将情感值+2。
Step6.根据分句情感值加和得到整句情感值。
Step7.将整句情感值进行处理。用积极值和消极值作差，得到的称作该句的情感分数，如为正值则输出“1”，如为负值则输出“-1”，如相等则输出“0”。
Step8.计算并记录所有评论的情感分数。

5.5.2 群众再访率

除了以上已实现的模型以及对情感分析的优化，我们认为还有一些地方可以进行优化。由于该附件仅有群众的一次留言及政府的答复，无法得知群众在看到答复后的情绪、意见，因此可以通过筛选出同一用户对同一问题，并计算分析其留言次数、情感，以及对比群众实际反馈与政府答复中的承诺，可以评价出群众对政府答复的满意度。但由于时间因素以及其他原因，我们并未对其进行程序实现，此后会进行相关方面的研究。

5.5.3 模型优化后的权重：

得分 = 可解释性（25%）+时效性（20%）+相关性（15%）+完整性（12%）+详细程度（12%）+情绪分析（8%）+群众再访率（8%）

这个权重也是采用了主观赋权法和客观赋权法相结合的方法进行确定的，其中相比于优化前的模型，我们降低了可解释性和相关性的权重，并且将其移动到了情绪分析上，这个移动其实比较主观。主要是在加权平均的过程中，如果添加的因素权重较低就会导致对整体情况的影响过小，同时情绪分析并不是对答复评价的主要因素，所以权重过高又会导致评价结果失真，综合考虑下这个比重比较合理。同理群众再访率也是我们考虑的一个没有那么多比重的一个因素，所以权重也比较低。

由于时间关系，我们并没有将最终优化好的模型的程序完整的写出来，但是我们写了情感分析的程序，需要使用时可以直接进行调用。

6 总结

综上所述，我们通过利用 python 中的数据挖掘相关的库和机器学习的知识，解决了题目中的三个具体问题，同时也根据题目建立了有关模型并对模型进行了评价和优化。在实现算法的过程中我们碰到了不少问题，且最后的结果仍有些不足。如问题一中的 f1-score 模型评价结果分值不算高等。在后面的工作中如果有机会我们会继续使用 python 及机器学习相关工具进行完善。

参考文献

[1] https://blog.csdn.net/FontThrone/article/details/82807816
[2] https://blog.csdn.net/wjg8209/article/details/104426971
[3] https://www.jianshu.com/p/d7e7cc747e56
[4] https://cloud.tencent.com/developer/article/1476898
[5]
[6]王宏勇. 网络舆情热点发现与分析研究[D].西南交通大学,2011.
[7]杨经. 网络舆情热点话题发现技术研究[D].福州大学,2011.