针对“智慧政务”中文本挖掘应用问题的研究

摘要

随着大数据、云计算、人工智能等技术的发展，建立基于自然语言处理技术的智慧政务系统对提升政府的管理水平和施政效率具有极大的推动作用。而本文主要运用 Jupyter notebook 和 Matlab 解决以下三个问题：群众留言分类、热点问题挖掘、答复意见的评价。

针对问题 1：本题需要建立关于留言内容的一级标签分类模型，并使用 F-score 对分类方法进行评价。由于一级分类标签的各个数目不相同，所以本文采用分层抽样的方法对数据进行抽取，使用朴素贝叶斯分类器、kNN 分类器和支持向量机分类器进行研究。通过实验最终发现 kNN 分类器构造的模型效果最差 F_1值仅仅为 0.50；而朴素贝叶斯分类器构造的模型的 F_1值为 0.87；支持向量机分类器构造的模型的 F_1值也为 0.87。

针对问题二：本题需要从众多留言中识别出相似的留言，并把相似的留言归为同一问题，然后进行热度评价。本文使用 K-means 聚类方法最终将问题聚为四类。通过归一化处理和熵权法进行热度评价并对热度指数进行排名，排名前五的热点问题分别为 A 市 A5 区汇金路五矿万境 K9 县小区群租房泛滥成灾、A 市金毛湾 2800 户业主担心的学位问题、A 市 A4 区 p2p 公司 58 车贷问题、A4 区绿地海外滩小区小区旁边建高铁铁路扰民、A9 市高铁站选址。

针对问题三，本题需要对答复意见的进行评价，根据相关性、完整性、可解释性，将三个指标进行量化处理，建立评价体系，最后对的质量进行综合评价。本文采用了 AHP-模糊综合评价方法，一级指标判断矩阵及层次计算的权向量为 W=[0.4,0.3,0.3],最终答复意见综合质量评价体系 G=1/3（G1+G2+G3）。

关键词：朴素贝叶斯 支持向量机 熵权法 AHP-模糊综合评价 文本挖掘

目录

一、问题重述.....3
二、 符号说明.....3
三、问题分析与求解.....4
3.1 针对问题一的分析与求解.....4
3.1.1 问题一的分析.....4
3.1.1 问题一的求解.....4
3.2 针对问题二的分析与求解.....10
3.2.1 问题二的分析.....10
3.2.1 问题二的求解.....10
3.3 针对问题三的分析与求解.....13
3.3.1 问题三的分析.....13
3.3.2 问题三的求解.....14
四、模型的评价.....18
4.1 模型的优点.....18
4.2 模型的缺点.....18
五、参考文献.....18
六、附录.....19
附录 1.....19
附录 2.....21
附录 3.....22

一、问题重述

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，各类社情民意相关的文本数据量不断攀升，给以往主要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。同时，随着大数据、云计算、人工智能等技术的发展，建立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。

问题一：群众留言分类。建立关于留言内容的一级标签分类模型。使用 F-Score 对分类方法进行评价。

问题二：热点问题挖掘。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，并给出评价结果，按表 1 的格式给出排名前 5 的热点问题。

问题三：答复意见的评价。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。

二、符号说明

序号	符号	说明
1	P	查准率
2	R	查全率
3	F1	F1系数
4	E_j	所有方案对属性的贡献总量
5	W_j	各属性权重
6	f_i	综合得分
7	A_{1i}	相关性指标
8	A_{2i}	完整性指标
9	A_{3i}	可解释性指标
10	W	权重
11	C.I	一致性指标
12	R.I	平均随机一致性指标
13	C.R	一致性比率
14	G	答复意见综合质量

三、问题分析与求解

3.1 针对问题一的分析与求解

3.1.1 问题一的分析

本题需要建立关于留言内容的一级标签分类模型。并使用 F-Score 对分类方法进行评价。常用的文本分类器有朴素贝叶斯、kNN、支持向量机、决策树和神经网络分类器。通过种种考虑最终本文决定选取朴素贝叶斯分类器、kNN 分类器和支持向量机分类器进行研究。

3.1.2 问题一的求解

1.数据读取：

将城乡建设设置为 1，将环境保护设置为 2，将交通运输设置为 3，将教育文体设置为 4，将劳动和社会保障设置为 5，将商贸旅游设置为 6，将卫生计生设置为 7。前五行的数据情况如下：

表 1 数据情况表

```
In [2]: data.head()
```

Out[2]:

	留言编号	留言用户	留言主题	留言时间	留言详情	一级标签	labels
0	24	A00074011	A市西湖建筑集团占道施工有安全隐患	2020/1/6 12:09:38	\n\t\t\t\t\t\n\t\t\t\t\t\tA3区大道西行便道，未管所路口至加油站路段，...	城乡建设	1
1	37	U0008473	A市在水一方大厦人为烂尾多年，安全隐患严重	2020/1/4 11:17:46	\n\t\t\t\t\t\n\t\t\t\t\t\t位于书院路主干道的在水一方大厦一楼至四楼人为...	城乡建设	1
2	83	A00063999	投诉A市A1区苑物业违规收停车费	2019/12/30 17:06:14	\n\t\t\t\t\t\n\t\t\t\t\t\t尊敬的领导：A1区苑小区位于A1区火炬路，小...	城乡建设	1
3	303	U0007137	A1区蔡锷南路A2区华庭楼顶水箱长年不洗	2019/12/6 14:40:14	\n\t\t\t\t\t\n\t\t\t\t\t\tA1区A2区华庭小区高层为二次供水，楼顶水箱...	城乡建设	1
4	319	U0007137	A1区A2区华庭自来水好大一股霉味	2019/12/5 11:17:22	\n\t\t\t\t\t\n\t\t\t\t\t\tA1区A2区华庭小区高层为二次供水，楼顶水箱...	城乡建设	1

根据输出结果可知共有 7 列数据分别是留言编号、留言用户、留言主题、留言时间、留言详情、一级标签和 labels。

2. 数据探索分析

统计一级标签中城乡建设（1）、环境保护（2）、交通运输（3）、教育文体（4）、劳动和社会保障（5）、商贸旅游（6）、卫生计生（7）各自的总数。

表 2 一级标签数量表

```
In [3]: data['一级标签'].value_counts()

Out[3]: 城乡建设        2009
        劳动和社会保障    1969
        教育文体        1589
        商贸旅游        1215
        环境保护         938
        卫生计生         877
        交通运输         613
        Name: 一级标签, dtype: int64
```

表 3 labels 数量表

```
In [4]: data['labels'].value_counts()

Out[4]: 1    2009
        5    1969
        4    1589
        6    1215
        2     938
        7     877
        3     613
        Name: labels, dtype: int64
```

由输出结果可知城乡建设（1）的数量最多，共有 2009 条，然后依次为 1969 条劳动和社会保障（5），1589 条教育文体（4），1215 条商贸旅游（6），938 条环境保护（2），877 条卫生计生（7），613 条交通运输（3）。

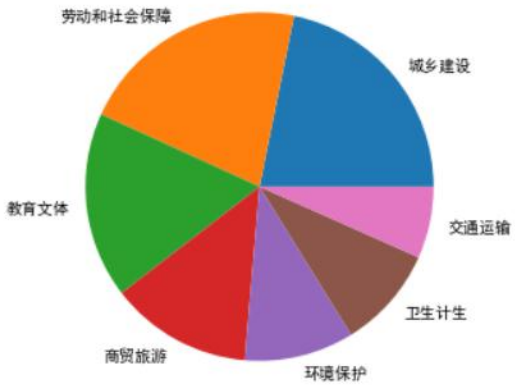

图 1 一级标签饼图

根据饼图可清楚的看出各个部分的大小及比份。

3.数据抽取

根据数据探索分析可知城乡建设（1）、环境保护（2）、交通运输（3）、教育文体（4）、劳动和社会保障（5）、商贸旅游（6）、卫生计生（7）的数据大小差异较大，为尽可能避免误差，本文采取分层抽样对每一个标签进行数据抽取并将所抽取的数据进行纵向拼接。

表 4 数据抽取表

```
In [57]: print(data_new.shape)
         print(data_new['labels'].value_counts())

         (4200, 7)
         7    600
         3    600
         6    600
         2    600
         5    600
         1    600
         4    600
         Name: labels, dtype: int64
```

根据输出结果可知，最终进行数据抽取、拼接后的数据共有 4200 行 7 列。其中城乡建设（1）、环境保护（2）、交通运输（3）、教育文体（4）、劳动和社会保障（5）、商贸旅游（6）、卫生计生（7）分别都为 600 条。完成了对不均衡的数据均衡化。

4.文本预处理

本文在进行文本预处理是先将不是汉字的符号进行了剔除。

5.分词和去除停用词

对分完词之后的数据进行去除停用词的操作，对列表里面的每一个词进行判断，判断该词有没有在上面的停用词表里面，如果不在就将它保留下来。

6.文本向量化表示

首先需要统计相应的词频去做向量框类似与词频向量的展示，由于 sklearn 库是 Google 开展出来的，所以针对的文本更多的是英文，所以对于中文上的数据是不支持的，通过空格将词连结，构建完词典后进行转换。

表 5 文本向量化表示表

```
In [63]: from sklearn.feature_extraction.text import CountVectorizer
         cv=CountVectorizer().fit(tmp)
         cv_data=cv.transform(tmp)
         cv_data.toarray()

Out[63]: array([[0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0]], dtype=int64)

In [64]: cv_data.toarray().shape

Out[64]: (4200, 50099)
```

运行结果以二维数组的形式将词频参数显示出来。每一个文本对应一个数据，将非结构化的数据转化为了结构化的数据。该数据共有 4200 行 50099 列，该矩阵为稀疏矩阵。

7.分类器

由于文本分类本身是一个分类问题，因此一般的模式分类方法都可以用于文本分类研究。常用的文本分类器有朴素贝叶斯、kNN、支持向量机、决策树和神经网络分类器。由于输出矩阵为稀疏矩阵所以在进行文本分类时在众多方法中我们排除神经网络进行分类，因为 NNet 分类器的效果比 kNN 分类器和 SVM 分类器的效果差，且训练时间开销远远超过其他分类器，因此本文不使用。本文选取朴素贝叶斯分类器、kNN 分类器和支持向量机分类器进行研究。

将数据进行切分，切分为训练集和测试集，按照百分比对训练集和测试集进行切分，其中百分之二十的数据为测试集，百分之八十为训练集。

方法一：朴素贝叶斯。本文主要采用的是多项式朴素贝叶斯。朴素贝叶斯算法流程如下所示：

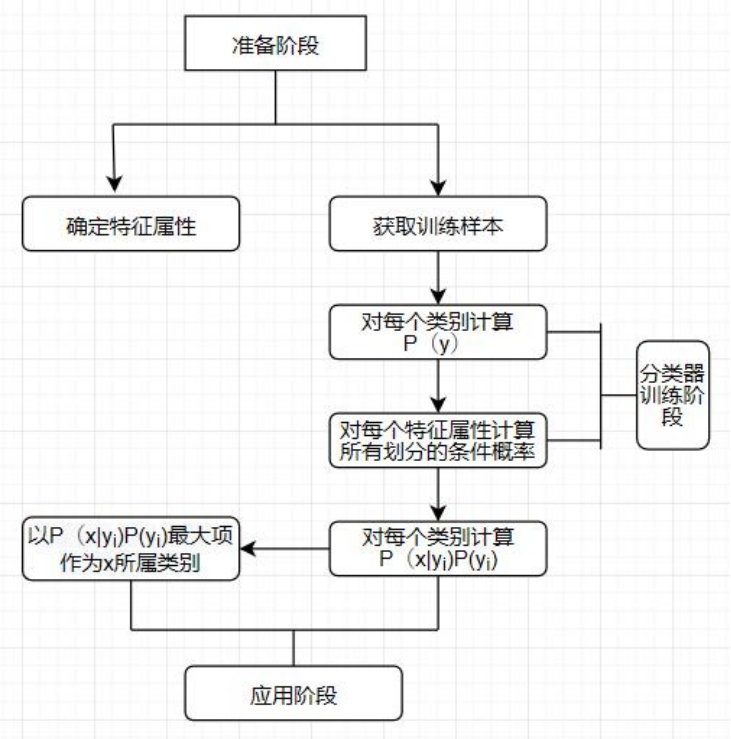

图 2 朴素贝叶斯算法流程图

表 6 朴素贝叶斯正确率表

```
In [67]: model_nb=MultinomialNB().fit(cv_train, y_train)
         model_nb.score(cv_test, y_test)

Out[67]: 0.8666666666666667
```

根据输出结果可知朴素贝叶斯的正确率为 0.87。

方法二：kNN。kNN 算法计算步骤如下:

1)算距离:给定测试对象，计算它与训练集中的每个对象的距离

2)找邻居:圈定距离最近的 k 个训练对象，作为测试对象的近邻

3)做分类:根据这 k 个近邻归属的主要类别，来对测试对象分类

kNN 是一种懒惰算法。懒惰的后果:构造模型很简单，但在对测试样本分类地的系统开销大，因为要扫描全部训练样本并计算距离。

表 7 kNN 正确率表

```
In [68]: model_knn=KNeighborsClassifier().fit(cv_train, y_train)
         model_knn.score(cv_test, y_test)

Out[68]: 0.46190476190476193
```

根据输出结果可知 kNN 的正确率为 0.46。

方法三：支持向量机。本文主要采用线性核函数。

表 8 支持向量机正确率表

```
In [69]: model_svc=LinearSVC().fit(cv_train, y_train)
         model_svc.score(cv_test, y_test)

         C:\ProgramData\Anaconda3\lib\site-packages\sklearn\svm\_base.py:947: ConvergenceWarnin
         g: Liblinear failed to converge, increase the number of iterations.
           "the number of iterations.", ConvergenceWarning)

Out[69]: 0.8595238095238096
```

根据输出结果可知支持向量机的正确率为 0.86。

8.模型评估

首先对 test 中的数据进行预测，预测结束后构建分类报告，从而可以得到查准率、查全率、正确率。

查准率：

$$P = \frac{TP}{TP + FP}$$

查全率：

$$R = \frac{TP}{TP + FN}$$

F_1 系数：

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

其中 P_i 为第 i 类的查准率，R_i 为第 i 类的查全率。

对于方法一朴素贝叶斯模型的评估：

表 9 朴素贝叶斯模型评估表

```
In [71]: y_pre_nb=model_nb.predict(cv_test)
         print(classification_report(y_true=y_test, y_pred=y_pre_nb))
         print(confusion_matrix(y_true=y_test, y_pred=y_pre_nb))
```

```
              precision    recall  f1-score   support

           1       0.78      0.75      0.76       111
           2       0.91      0.98      0.94       136
           3       0.86      0.86      0.86       110
           4       0.92      0.93      0.92       130
           5       0.78      0.89      0.83       101
           6       0.87      0.79      0.83       119
           7       0.93      0.84      0.88       133

    accuracy                           0.87       840
   macro avg       0.86      0.86      0.86       840
weighted avg       0.87      0.87      0.87       840

[[ 83   8   7   3   4   4   2]
 [  2 133   0   0   0   1   0]
 [ 10   1  95   1   2   1   0]
 [  0   1   0 121   6   2   0]
 [  0   0   0   4  90   1   6]
 [ 10   1   8   3   2  94   1]
 [  2   2   1   0  11   5 112]]
```

根据输出结果可知该模型的查准率为 0.87，查全率为 0.87，F_1 值为 0.87。

对于方法二 kNN 模型的评估：

表 10 kNN 模型评估表

```
In [72]: y_pre_knn=model_knn.predict(cv_test)
         print(classification_report(y_true=y_test, y_pred=y_pre_knn))
         print(confusion_matrix(y_true=y_test, y_pred=y_pre_knn))
```

```
              precision    recall  f1-score   support

           1       0.57      0.32      0.41       111
           2       0.89      0.35      0.51       136
           3       0.69      0.38      0.49       110
           4       0.85      0.47      0.60       130
           5       0.65      0.47      0.54       101
           6       0.22      0.83      0.34       119
           7       0.86      0.42      0.57       133

    accuracy                           0.46       840
   macro avg       0.68      0.46      0.49       840
weighted avg       0.69      0.46      0.50       840

[[35  2  1  1  2 67  3]
 [10 48  3  1  2 72  0]
 [ 3  0 42  0  2 62  1]
 [ 6  2  3 61  4 52  2]
 [ 3  0  3  6 47 40  2]
 [ 3  1  7  2  6 99  1]
 [ 1  1  2  1  9 63 56]]
```

根据输出结果可知该模型的查准率为 0.69，查全率为 0.46，F_1 值为 0.50。对于方法三支持向量机模型的评估：

表 11 支持向量机模型评估表

```
In [73]: y_pre_svm=model_svc.predict(cv_test)
         print(classification_report(y_true=y_test, y_pred=y_pre_svm))
         print(confusion_matrix(y_true=y_test, y_pred=y_pre_svm))
```

```
              precision    recall  f1-score   support

           1       0.72      0.74      0.73       111
           2       0.93      0.91      0.92       136
           3       0.83      0.84      0.83       110
           4       0.93      0.93      0.93       130
           5       0.83      0.89      0.86       101
           6       0.81      0.81      0.81       119
           7       0.94      0.88      0.91       133

    accuracy                           0.86       840
   macro avg       0.86      0.86      0.86       840
weighted avg       0.86      0.86      0.86       840

[[ 82   6   7   2   4   8   2]
 [  7 124   0   0   2   3   0]
 [ 14   0  92   0   0   4   0]
 [  1   0   1 121   4   3   0]
 [  3   0   1   3  90   0   4]
 [  5   2  10   4   1  96   1]
 [  2   1   0   0   8   5 117]]
```

根据输出结果可知该模型的查准率为 0.86，查全率为 0.86，F_1 值为 0.86。

综合三种模型来看，kNN 的效果比较差，在选择模型时一般要求查准率更高一点，查准率越高越好。根据三个模型的查准率、查全率和 F_1 值的综合考虑认为朴素贝叶斯进行文本分类的模型最优。

3.2 针对问题二的分析与求解

3.2.1 问题二的分析

本题需要从众多留言中识别出相似的留言，并把相似的留言归为同一问题，然后进行热度评价。经过反复测试与筛选，最终选择了 jieba 库、TF-IDF 算法和 K-means 聚类来实现问题分类；采用归一化处理、熵权法来进行热度评价。

3.2.2 问题二的求解

1.数据读取

前五行的数据情况如下：

表 12 数据情况表

```
In [1]: import pandas as pd
        data=pd.read_csv('bisai/C题全部数据/C题全部数据/附件3.csv',encoding='GBK')

In [2]: data.head()
```

Out[2]:

	留言编号	留言用户	留言主题	留言时间	留言详情	反对数	点赞数
0	188006	A000102948	A3区一米阳光婚纱艺术摄影是否合法纳税了?	2019/2/28 11:25:05	\n\t\t\t\t\t\n\t\t\t\t\t座落在A市A3区联丰路米兰春天G2栋320，...	0	0
1	188007	A00074795	咨询A6区道路命名规划初步成果公示和城乡门牌问题	2019/2/14 20:00:00	\n\t\t\t\t\t\n\t\t\t\t\tA市A6区道路命名规划已经初步成果公示文件，...	0	1
2	188031	A00040066	反映A7县春华镇金鼎村水泥路、自来水到户的问题	2019/7/19 18:19:54	\n\t\t\t\t\t\n\t\t\t\t\t本人系春华镇金鼎村七里组村民，不知是否有相关...	0	1
3	188039	A00081379	A2区黄兴路步行街大古道巷住户卫生间粪便外排	2019/8/19 11:48:23	\n\t\t\t\t\t\n\t\t\t\t\t靠近黄兴路步行街，城南路街道、大古道巷、一步...	0	1
4	188059	A00028571	A市A3区中海国际社区三期与四期中间空地夜间施工噪音扰民	2019/11/22 16:54:42	\n\t\t\t\t\t\n\t\t\t\t\tA市A3区中海国际社区三期四期中间，即蓝天璞...	0	0

根据输出结果可知共有 7 列数据分别是留言编号、留言用户、留言主题、留言时间、留言详情、反对数和点赞数。

2. 文本预处理

使用 jieba 分词组件来进行文本预处理，载入停用词表的词典，对文本实现去停用词，分词的操作，然后进行关键词的提取。

表 13 关键词提取表

```
In [3]: import re

In [4]: tmp=data['留言主题'].apply(lambda x:re.sub('[^\u4E00-\u9FD5]+','',x))

In [5]: import jieba

In [6]: with open('bisai/停用词表.txt','r',encoding='utf-8')as f:
            stop=f.read()
        stop=stop.split()
        stop=['']+stop

In [7]: data_cut=tmp.apply(jieba.lcut)
        data_cut.apply(lambda x:[i for i in x if i not in stop])
```

```
Building prefix dict from the default dictionary ...
Loading model from cache C:\Users\Dell\AppData\Local\Temp\jieba.cache
Loading model cost 1.026 seconds.
Prefix dict has been built successfully.
```

Out[7]:

```
0                          [区, 一米阳光, 婚纱, 艺术摄影, 合法, 纳税]
1              [咨询, 区, 道路, 命名, 规划, 初步, 成果, 公示, 城乡, 门牌]
2                    [县, 春华, 镇金鼎村, 水泥路, 自来水, 到户]
3                [区, 黄兴路, 步行街, 古道, 巷, 住户, 卫生间, 粪便, 外排]
4         [市区, 中海, 国际, 社区, 三期, 四期, 空地, 夜间, 施工, 噪音, 扰民]
                                ...
4321               [市, 经济, 学院, 寒假, 过年, 期间, 组织, 学生, 工厂]
4322                     [市, 经济, 学院, 组织, 学生, 外出, 打工]
4323                        [市, 经济, 学院, 强制, 学生, 实习]
4324                     [市, 经济, 学院, 强制, 学生, 外出, 实习]
4325                  [市, 经济, 学院, 体育, 学院, 变相, 强制, 实习]
Name: 留言主题, Length: 4326, dtype: object
```

运行结果把留言主题里面的关键词提取出来。

3. 文本向量化表示

文本表示是自然语言处理中的基础工作，文本表示的好坏直接影响到整个自然语言处理系统的性能。文本向量化就是将文本表示成一系列能够表达文本语义

的向量，是文本表示的一种重要方式。

表 14 文本向量化表示表

```
In [8]: data_after=data_cut.apply(
            lambda x:[i for i in x if i not in stop]
        )

In [9]: tmp=data_after.apply(lambda x:' '.join(x))

In [10]: from sklearn.feature_extraction.text import CountVectorizer
         cv=CountVectorizer().fit(tmp)
         cv_data=cv.transform(tmp)
         cv_data.toarray()

Out[10]: array([[0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

运行结果将非结构化的数据转化为了结构化的数据。

4.k-means 算法

k-means 算法属于无监督学习的一种聚类算法，在不知道数据所属类别及数量的前提下，依据数据自身所暗含的特点对数据进行聚类。

表 15 聚类结果表

```
In [14]: kmean_labels
```

Out[14]:

	留言主题	留言时间	留言详情
0	A3区一米阳光婚纱艺术摄影是否合法纳税了?	2019/2/28 11:25:05	0
1	咨询A6区道路命名规划初步成果公示和城乡门牌问题	2019/2/14 20:00:00	3
2	反映A7县春华镇金鼎村水泥路、自来水到户的问题	2019/7/19 18:19:54	0
3	A2区黄兴路步行街大古道巷住户卫生间粪便外排	2019/8/19 11:48:23	0
4	A市A3区中海国际社区三期与四期中间空地夜间施工噪音扰民	2019/11/22 16:54:42	1
...	...	...	...
4321	A市经济学院寒假过年期间组织学生去工厂工作	2019-11-22 14:42:14	0
4322	A市经济学院组织学生外出打工合理吗?	2019-11-05 10:31:38	0
4323	A市经济学院强制学生实习	2019-04-28 17:32:51	0
4324	A市经济学院强制学生外出实习	2018-05-17 08:32:04	0
4325	A市经济学院体育学院变相强制实习	2017-06-08 17:31:20	0

4326 rows × 3 columns

根据运行结果可知 k-means 聚类法将某一时段内反映特定地点或特定人群问题的留言一共聚为了四类。

5.数据预处理

先将附件三中的点赞数和反对数进行数据处理，选择归一化处理方法，采用了以下公式：

$$y_{ij}=\frac{x_{ij}-x_{jm}}{x_{jM}-x_{jm}},$$

经过上面的变换，所有的数据都变到[0，1]，便于后续工作进行统一处理

6.熵权法

对于归一化处理后的数据，对其指标的信息熵进行计算，采用以下的公式：

$$E_j = -\frac{\sum_{i=1}^{n} p_{ij} \ln p_{ij}}{\ln n},$$

$$p_{ij} = \frac{y_{ij}}{\sum_{j=1}^{m} y_{ij}}$$

其中

再计算客观权重，采用以下公式：

$$W_j = \frac{1 - E_j}{m - \sum_{j=1}^{m} E_j}$$

7.综合排名

采用归一化处理后的数据，采用熵权法的客观权重，对各个指标进行加权平均，可以得到各个指标的综合得分，计算式为：

$$f_1 = \sum_{j=1}^{m} w_j . y_{ij}$$

得到了以下（排名前五）的结果：

表 16 热度排名表

热度排名	问题ID	热度指数	留言时间	地点人群	问题描述
1	1	0.134079	2019/8/19 11:34	A市A5区汇金路五矿万境K9县	小区群租房泛滥成灾
2	2	0.114621	2019/4/11 21:02	A市金毛湾	金毛湾2800户业主担心的学位问题
3	3	0.052494	2019/2/21 18:45	A市A4区	p2p公司58车贷问题
4	4	0.042775	2019/9/5 13:06	A4区绿地海外滩小区	小区旁边建高铁铁路扰民
5	5	0.021424	2019/8/1 13:48	A9市高铁站	A9市高铁站选址

3.3 针对问题三的分析与求解

3.3.1 问题三的分析

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

根据题干给出的文本数据与答复意见满意度的相关指标进行分类，为了便于对答复意见进行更加全面、准确、定性的综合评价，本文采用层次分析法和模糊综合评价法相结合的方法，即模糊层次分析法，分层次进行模糊综合评价。

3.3.2 问题三的求解

1.层次结构模型

答复意见的质量的满意度情况是指相关部门对问题答复的与相关性、完整性、可解释性的一个综合评价，并将三种二级指标细分为：关键词的契合度 A11、问题的概括度 A21、答复的规范度 A22、问题的针对度 A31、答复的准确度 A32 五个三级指标，要建立一个系统的评价体系，根据以上三种印象因素进行分类。根据答复意见满意度的情况，初步建立答复意见满意度评价层次模型，如表 17。

表 17 答复意见质量评价层次模型

第一指标	第二指标	第三指标
答复意见质量	相关性 A1	关键词的契合度 A11
		关键句的契合度 A12
		关键段的切合度 A13
	完整性 A2	问题的概括度 A21
		答复的规范度 A22
		内容的充分度 A23
	可解释性 A3	问题的针对度 A31
		答复的准确度 A32
		建议的可行度 A33

2.答复意见质量指标量化分析

由于答复意见的指标类型具有多样性，对于划分答复意见质量等级的程度不亚于，需要所以对各类自然灾害划分评定指标，构建最底层的模糊矩阵。

表 18 指标赋值统计结果

第一指标	第二指标	第三指标	好	一般	差
答复意见质量	相关性 A_1	关键词的契合度 A_{11}	0.6	0.25	0.15
		关键句的契合度 A_{12}	0.45	0.35	0.2
		关键段的切合度 A_{13}	0.5	0.25	0.25
	完整性 A_2	问题的概括度 A_{21}	0.5	0.25	0.25
		答复的规范度 A_{22}	0.5	0.35	0.15
		内容的充分度 A_{23}	0.45	0.35	0.2

	可解释性 A_3	问题的针对度 A_{31}	0.5	0.35	0.15
		答复的准确度 A_{32}	0.45	0.4	0.15
		建议的可行度 A_{33}	0.5	0.35	0.15

3.指标权重求解的模糊层次分析法步骤

首先，根据 Satty 提供的重要性标度参考表，对表 1 的前三个指标进行打分， 然后建立判断矩阵，再运用规依法计算相应矩阵的权重，在进行一致性检验。本文采用对 n 个列向量求取平均值作为最后的权重。

公式：

$$W_i = \frac{1}{n}\sum_{j=1}^{n}\frac{a_{ij}}{\sum_{k=1}^{n}a_{ki}}$$

根据表的答复意见评价体系以及各个因素所占比的权重，将同一级指标进行两两比较，得出每一级指标的判断矩阵，并计算各级对应目标层的权重。用 MATLAB 求解，计算如下表所示：

表 19 一级指标判断矩阵及层次计算权向量

层次 A	A_1	A_2	A_3	W
A_1	1	$\frac{4}{3}$	$\frac{4}{3}$	0.4
A_2	$\frac{3}{4}$	1	1	0.3
A_3	$\frac{3}{4}$	1	1	0.3

根据上表得到一级指标判断矩阵及层次计算的权向量 W=[0.4,0.3,0.3]用上述构建判断矩阵和求层次向量权重的方法，同样可以求解二级指标 A1、A2、A3、A4 的判断矩阵和权向量，用 MATLAB 求解，计算结果如表所示。

表 20 二级指标判断矩阵及层次计算权向量

层次 A	A_{11}	A_{12}	A_{13}	W_1
A_{11}	1	$\frac{10}{7}$	$\frac{10}{3}$	0.5
A_{12}	$\frac{7}{10}$	1	$\frac{7}{3}$	0.35
A_{13}	$\frac{3}{10}$	$\frac{3}{7}$	1	0.15
层次 A	A_{21}	A_{22}	A_{23}	W_2
A_{21}	1	$\frac{5}{3}$	$\frac{5}{2}$	0.5
A_{22}	$\frac{3}{5}$	1	$\frac{3}{2}$	0.3
A_{23}	$\frac{2}{5}$	$\frac{2}{3}$	1	0.2
层次 A	A_{31}	A_{32}	A_{33}	W_3
A_{31}	1	$\frac{3}{2}$	$\frac{9}{5}$	0.45
A_{32}	$\frac{2}{3}$	1	$\frac{6}{5}$	0.3
A_{33}	$\frac{5}{9}$	$\frac{5}{6}$	1	0.25

根据上表得到二级指标判断矩阵及层次计算权向量分别为 W1=[0.5,0.35,0.15],W2=[0.5,0.3,0.2],W3=[0.45,0.3,0.25]。

4.判断矩阵的一致性检验

在实际应用中,要求判断矩阵满足大体上的一致性,因此需进行一致性检验。首

先计算一致性指标 C.I.采用公式$C.I=\frac{\lambda_{\max}-n}{n-1}$，，然后查表确定相应的平均随机一致 性指标 R.I.最后，计算比例 C.R.并进行判断，$C.R=\frac{C.I}{R.I}$,当 C.R.<0.1 时，认为判 断矩阵的一致性是可以接受的，当 C.R.>0.1 时，认为判断矩阵不符合一致性要求， 需要对该判断矩阵重新修正。为了操作的便捷性及科学性，这一步用 MATLAB 进行，计算结果如表：

表 21 判断矩阵一致性检验

判断矩阵	A	A1	A2	A3
一致性指标 C.I	$-6.6613*10^{-16}$	$2.2204*10^{16}$	$-8.8818*10^{-16}$	$-2.2204*10^{-16}$
一致性比例 C.R	$-1.1485*10^{-15}$	$3.8284*10^{-16}$	$1.5313*10^{-15}$	$-3.8284*10^{-16}$
一致性检查结果	通过	通过	通过	通过

根据一致性检验结果可知全部指标均通过一致性检验。

5.答复质量的评价体系

答复质量的衡量标准依据三个准则：（1）一致性，对于意见的答复，需要和问题有极高的契合度；

（2）完整性，对于意见的答复，需要对问题具有高度的概括，以及满足规范性；

（3）可解释性，对于意见的答复，需要对问题具有针对性，不可答非所问，并且需要有极高的准确性。本文采用以下方式进行定量评估。

G 代表答复综合质量，以下有：

（1）相关性 A_1

（2）完整性 A_2：问题的概括度 A_{21}；答复的规范度 A_{22}

（3）可解释性 A_3 :问题的针对度 A_{31}；答复的准确度 A_{32}

得到如下结果：

G1=1/3（0.5A11+0.35A12+0.15A13）

G2=1/3（0.5A21+0.3A22+0.2A23）

G3=1/3（0.45A31+0.3A32+0.25A33）

G=1/3（G1+G2+G3）

6.答复质量划分

利用上诉公式计算出答复质量 G，能够对各个答复信息进行评估。本位将答复质量划分为三个等级，划分所遵循的准则：（1）客观性，对于划分评价等级，不能带有主观性，不能盲目划分；（2）普遍性，由于答复样式过多，对于划分评价等级，需要“大众化”，不能取极端例子；（3）实际性，对于划分评价等级，要结合实际。本文对答复质量的划分：

表 22 质量得分

答复质量划分	差	一般	好
答复质量	≤60	≤80	≤100

四、模型的评价

4.1 模型的优点

本文的模型主要采用的分层抽样的方法抽取数据，将原始不均衡的数据变为均衡的数据，分层抽样可以提高总体参数估计的精度使实验更加科学。使用朴素贝叶斯分类器、kNN 分类器和支持向量机分类器分别进行研究然后进行对比，分类器的准确度高，对数据没有假设使实验结果更具有说服性。

层次分析是一种系统性的分析法，将研究对象作为一个系统，按照分解、比较判断、综合的思维方式进行决策，并且简洁实用，所需定量数据信息较少。模糊综合预测模型是在模糊环境下,考虑多种因素的影响,为了某种目的对一事物做出综合决策的方法。

4.2 模型的缺点

为了运行时间较短，抽取数据相对于原数据来说较少，因此在一定程度上降低了准确度。kNN 算法计算量大，需要大量的内存，输出的可解释性不强。层次分析法指标过多时，数据统计量大，且权重难以确定，特征值和特征向量的精确求法比较复杂。综合模糊评价模型的缺点在于对指标的定性过于模糊时，评价的最终结果与事实存在偏差。

五、参考文献

[1]嵩天,黄天羽,礼欣.Python 语言:程序设计课程教学改革的理想选择[J].中国大学教学, 2016(2):42-47.

[2]车万翔,苏小红,袁永峰叶麟.计算机专业高级语言程序设计课程改革探索[J].计算机教育,2014(13):56-63.

[3] (丘恩 Chun) (W.J.) . Python 核心编程(第 2 版) [M]. 人民邮电出版社.2008.

[4] 张若愚 Python 科学计算[M]:清华大学出版社，2012.

[5] 巴里深入浅出 Python[M].北京: 东南大学出版社，2011.

六、附录

附录 1

问题 1 代码：

```
import pandas as pd
data=pd.read_csv('bisai/C 题全部数据/C 题全部数据/附件2.1.csv',encoding='GBK')
data.head()
data['一级标签'].value_counts()
data['labels'].value_counts()
num=data['labels'].value_counts()
import matplotlib.pyplot as plt
plt.rcParams['font.sans-serif']='SimHei'
plt.figure(figsize=(5, 5))
plt.pie(num, labels=['城乡建设','劳动和社会保障','教育文体','商贸旅游','环境保护','卫生计生','交通运输'])
plt.show()
data1=data.loc[data['labels']==1,].sample(100,random_state=123)
data2=data.loc[data['labels']==2,].sample(100,random_state=123)
data3=data.loc[data['labels']==3,].sample(100,random_state=123)
data4=data.loc[data['labels']==4,].sample(100,random_state=123)
data5=data.loc[data['labels']==5,].sample(100,random_state=123)
data6=data.loc[data['labels']==6,].sample(100,random_state=123)
data7=data.loc[data['labels']==7,].sample(100,random_state=123)
data_new=pd.concat([data1,data2,data3,data4,data5,data6,data7])
data_new.shape
data_new['labels'].value_counts()
print(data_new.shape)
print(data_new['labels'].value_counts())
import re
```

```
    tmp=data_new['      留      言      详      情      '].apply(lambda
x:re.sub('[^\u4E00-\u9FD5]+','',x))
    import jieba
    with open('bisai/停用词表.txt','r',encoding='utf-8')as f:
        stop=f.read()
    stop=stop.split()
    stop=['']+stop
    data_cut=tmp.apply(jieba.lcut)
    data_cut.apply(lambda x:[i for i in x if i not in stop])
    data_after=data_cut.apply(
        lambda x:[i for i in x if i not in stop]
    )
    tmp=data_after.apply(lambda x:' '.join(x))
    from sklearn.feature_extraction.text import CountVectorizer
    cv.fit(tmp)
    cv_data=cv.transform(tmp)
    cv_data.toarray()
    cv_data.toarray().shape
    from sklearn.naive_bayes import MultinomialNB
    from sklearn.neighbors import KNeighborsClassifier
    from sklearn.svm import LinearSVC
    from sklearn.model_selection import train_test_split
    cv_train, cv_test, y_train, y_test= train_test_split(
        cv_data, data_new['labels'],
        test_size=0.2, random_state=123
    )
    model_nb=MultinomialNB().fit(cv_train, y_train)
    model_nb.score(cv_test, y_test)
    model_knn=KNeighborsClassifier().fit(cv_train, y_train)
```

```
model_knn.score(cv_test, y_test)
model_svc=LinearSVC().fit(cv_train, y_train)
model_svc.score(cv_test, y_test)
from sklearn.metrics import classification_report, confusion_matrix
y_pre_nb=model_nb.predict(cv_test)
print(classification_report(y_true=y_test, y_pred=y_pre_nb))
print(confusion_matrix(y_true=y_test, y_pred=y_pre_nb))
y_pre_knn=model_knn.predict(cv_test)
print(classification_report(y_true=y_test, y_pred=y_pre_knn))
print(confusion_matrix(y_true=y_test, y_pred=y_pre_knn))
y_pre_svm=model_svc.predict(cv_test)
print(classification_report(y_true=y_test, y_pred=y_pre_svm))
print(confusion_matrix(y_true=y_test, y_pred=y_pre_svm))
```

附录 2

问题 2 代码：

```
import pandas as pd
data=pd.read_csv('bisai/C 题全部数据/C 题全部数据/附件3.csv',encoding='GBK')
data.head()
import re
tmp=data['留言主题'].apply(lambda x:re.sub('[^\u4E00-\u9FD5]+','',x))
import jieba
with open('bisai/停用词表.txt','r',encoding='utf-8')as f:
    stop=f.read()
stop=stop.split()
stop=['']+stop
data_cut=tmp.apply(jieba.lcut)
data_cut.apply(lambda x:[i for i in x if i not in stop])
```

```
data_after=data_cut.apply(
    lambda x:[i for i in x if i not in stop]
)
tmp=data_after.apply(lambda x:' '.join(x))
from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer().fit(tmp)
cv_data=cv.transform(tmp)
cv_data.toarray()
from sklearn.cluster import KMeans
kmeans=KMeans(n_clusters=4,random_state=123).fit(cv_data)
kmean_labels=data[["留言主题","留言时间"]]
kmean_labels["留言详情"]=kmeans.labels_
kmean_labels
```

附录 3

层次分析法 matlab 代码：

```
disp('请输入判断矩阵 A(n 阶)');
A=input('A=');
[n,n]=size(A);
x=ones(n,100);
y=ones(n,100);
m=zeros(1,100);
m(1)=max(x(:,1));
y(:,1)=x(:,1);
x(:,2)=A*y(:,1);
m(2)=max(x(:,2));
y(:,2)=x(:,2)/m(2);
p=0.0001;i=2;k=abs(m(2)-m(1));
while    k>p
    i=i+1;
    x(:,i)=A*y(:,i-1);
    m(i)=max(x(:,i));
    y(:,i)=x(:,i)/m(i);
    k=abs(m(i)-m(i-1));
end
a=sum(y(:,i));
w=y(:,i)/a;
t=m(i);
```

```
disp('权重');disp(w);
disp('最大特征值');disp(t);
                %以下是一致性检验
CI=(t-n)/(n-1);RI=[0 0 0.52 0.89 1.12 1.26 1.36 1.41 1.46 1.49 1.52 1.54
1.56 1.58 1.59];
CR=CI/RI(n);
if CR<0.10
      disp('判断矩阵的一致性可以接受');
      disp('CI=');disp(CI);
      disp('CR=');disp(CR);
else
      disp('判断矩阵的一致性不可以接受');
end
```