第八届“泰迪杯”全国数据挖掘挑战赛论文报告

所选题目：C 题 ：

“智慧政务”中的文本挖掘应用

基于文本挖掘的分类与聚类技术应用

综合评定成绩：_____

评委评语：

评委签名：

基于文本挖掘的分类与聚类技术应用

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。

以大数据提升政府治理能力是大势所趋。科技革命的加速推进特别是大数据时代的到来，以大数据提升政府治理能力的第一步，迫切要求政府治理加快，大数据已成为“提升政府治理能力的新途径”。

通过云计算、人工智能等技术对留言的管理和挖掘的程序逐步的形成，留言管理挖掘程序是指通过创建完善的程序对留言进行分类和对热点问题进行挖掘，实现非人工留言分类，并根据不同种类的留言的数量进行统计来挖掘热点问题。

这就要求各级政府树立大数据思维，借助大数据手段推动政府管理理念和社会治理模式进步，实现国家治理体系和治理能力现代化。留言管理挖掘程序大大提高留言分类的效率，及时反馈信息，分类的准确性也更高。如何对信息快速分类，让信息更加对号入座，及时挖掘出有针对性的新消息，是我们研究的新方向。

本次的实验在 Windows10，**Anaconda** 里的 Spyder 的运行环境中完成，我们通过 **Spyder** 进行代码编写，大大提高我们对留言分类的数据分析效率。此外，我们采用**降维相关**算法处理数据集，对特征重要性进行排序，取重要性大的特征,分别使用 **pandas**、**jieba**、**re** 和 **sklearn** 进行数据优化及建模。pandas 纳入了大量库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具，Jieba 分词算法使用了基于前缀词典实现高效的词图扫描，同时利用 re.sub 使用函数对匹配项的替换进行复杂的处理,其中用到 sklearn 的机器学习的方式有:Classification 分类，Preprocessing 数据预处理。

对于这次的建模我们用到两个模型，一个是高斯朴素贝叶斯模型，我们选择使用高斯朴素贝叶斯，将数据集拆分为训练集和测试集。另一个是 OvR 模型，所得概率最高的那个模型对应的样本类型即认为是该预测样本的类型。

根据目标问题，我们对所给数据做了挖掘分析以及预处理和筛选，过滤掉异常数据，以便于之后的数学模型的建立以及题目的解答。

关键词：文本挖掘；Python；pandas；分类与聚类；Jieba 分词；特征降维；正则表达式；高斯朴素贝叶斯

Application of classification and clustering technology based on Text Mining

Abstract

With the development of big data, cloud computing, artificial intelligence and other technologies, the establishment of smart government system based on natural language processing technology has become a new trend of social governance innovation and development, which has a great role in promoting the management level and efficiency of the government.

It is the general trend to improve the governance ability of the government with big data. The acceleration of the scientific and technological revolution, especially the arrival of the era of big data, is the first step to improve the government's governance ability with big data, which has become a "new way to improve the government's governance ability".

Through cloud computing, artificial intelligence and other technologies, the management and mining procedures for messages are gradually formed. The message management mining procedure refers to the classification of messages and the mining of hot issues through the creation of a perfect program, the realization of non artificial message classification, and the mining of hot issues according to the number of different types of messages.

This requires governments at all levels to establish big data thinking, promote the progress of government management concept and social governance mode by means of big data, and realize the modernization of national governance system and governance capacity. Message management mining program greatly improves the efficiency of message classification, timely feedback information, classification accuracy is also higher. How to classify the information quickly, make the information more correct, and find out the new information in time is our new research direction.

This experiment is completed in the running environment of Spyder in Windows 10 and anaconda. We code through Spyder, which greatly improves the efficiency of data analysis of message classification. In addition, we use dimension reduction correlation algorithm to process data sets, sort the importance of features, take the features of great importance, and use pandas,

Jieba, re and sklearn to optimize and model the data respectively. Pandas includes a large number of databases and some standard data models, and provides the tools needed for efficient operation of large data sets. Jieba word segmentation algorithm uses prefix dictionary to achieve efficient word map scanning, and uses re.sub to perform complex processing on the replacement of matching items. Among them, machine learning methods using sklearn include classification, preprocessing Data preprocessing.

For this modeling, we use two models, one is Gaussian naive Bayes model. We choose to use Gaussian naive Bayes to divide the data set into training set and test set. The other is the ovr model. The sample type corresponding to the model with the highest probability is considered as the type of the prediction sample.

According to the target problem, we do the mining analysis, preprocessing and filtering of the given data, filter out the abnormal data, so as to facilitate the establishment of the mathematical model and the solution of the problem.

Keywords: Text mining; Python; pandas; classification and clustering; Jieba segmentation; feature dimensionality reduction; regular expression; Gaussian naive Bayes

目　　录

1. 挖掘目标

1.1 问题背景

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。

通过云计算、人工智能等技术对留言的管理和挖掘的程序逐步的形成，留言管理挖掘程序是指通过创建完善的程序对留言进行分类和对热点问题进行挖掘，实现非人工留言分类，并根据不同种类的留言的数量进行统计来挖掘热点问题。

以大数据助推政务管理精准化，面对越来越复杂多变的社会事务，政府应借助大数据手段进行政务管理，准确把握、及时发现问题，从而进一步提升政府监督管理的精准性和针对性。同时，通过深度数据挖掘分析，实时响应、处理公共事件和公众诉求。

1.2 目标任务

题目所给的附件 2、附件 3、附件 4 的数据来源于互联网公开渠道，团队需充分利用题目中给出的数据。

问题一：群众留言分类

在处理网络问政平台的群众留言时，工作人员首先按照一定的划分体系（参考附件 1 提供的内容分类三级标签体系）对留言进行分类，以便后续将群众留言分派至相应的职能部门 处理。目前，大部分电子政务

系统还是依靠人工根据经验处理，存在工作量大、效率低，且差错率高等问题。请根据附件 2 给出的数据，建立关于留言内容的一级标签分类模型。

问题二：热点问题挖掘

某一时段内群众集中反映的某一问题可称为热点问题，如“XXX 小区多位业主多次反映入夏以来小区楼下烧烤店深夜经营导致噪音和油烟扰民”。及时发现热点问题，有助于相关部门进行有针对性地处理，提升服务效率。请根据附件 3 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，按表 2 的格式给出相应热点问题对应的留言信息。

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。

2. 分析方法与过程

2.1 问题分析

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。于是，通过云计算、人工智能等技术对留言的管理和挖掘的程序逐步的形成，留言管理挖掘程序是指通过创建完善的程序

对留言进行分类和对热点问题进行挖掘，实现非人工留言分类，并根据不同种类的留言的数量进行统计来挖掘热点问题。留言管理挖掘程序大大提高留言分类的效率，及时反馈信息，分类的准确性也更高。如何对信息快速分类，让信息更加对号入座，及时挖掘出有针对性的新消息，是我们研究的新方向。因此：

针对问题一，我们结合赛方提供数据的实际情况，对留言的七个分类模块进行提取、合并，且对七个分类模块所给不同的数据量进行数量提取相同化。根据附件 2 给出的数据，建立关于留言内容的一级标签分类模型。

针对问题二，根据附件 3 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，我们给出合理的热度评价指标。

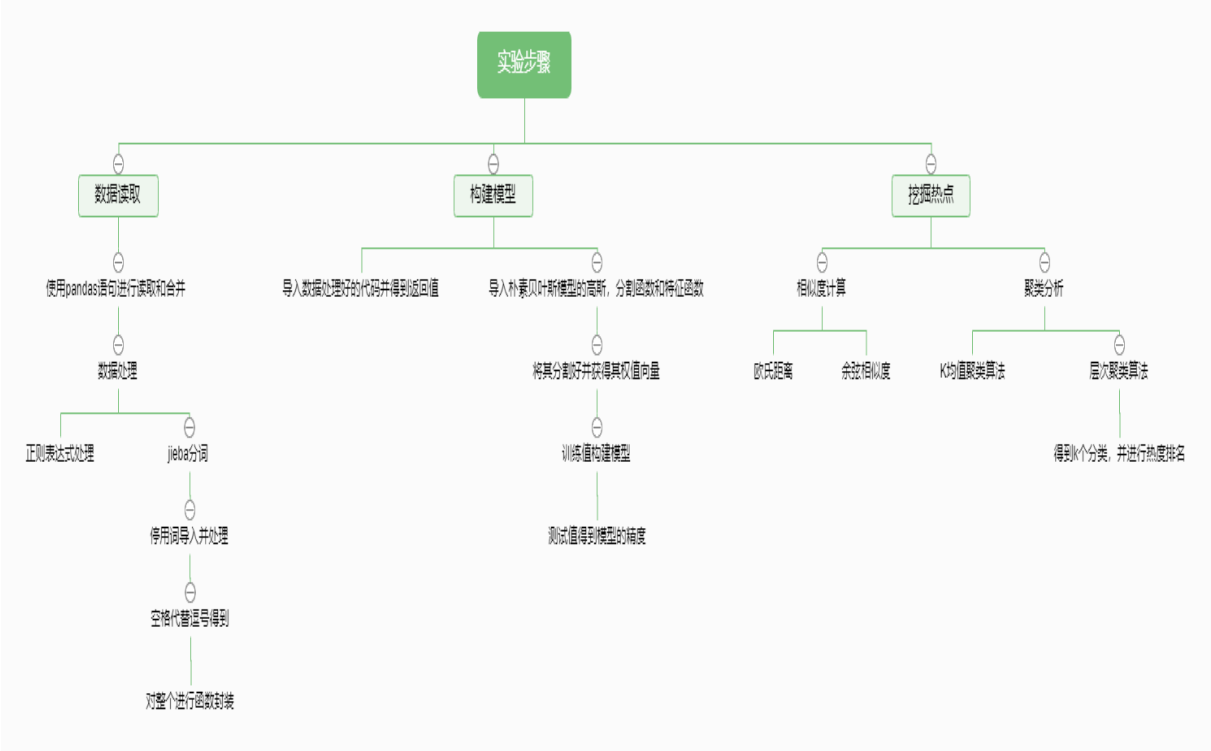
针对问题三，根据附件 4 相关部门对留言的答复意见，从答复的相关性、完整性、可解释性等角度对答复意见的质量，我们给出一套评价方案。

2.2 实验平台

本次的实验在 Windows10，Anaconda 里的 Spyder 的运行环境中完成。Anaconda 是一个开源的 Python 发行版本，能高性能使用 Python 语言，其包含了 conda、Python 等 180 多个科学包及其依赖项，是一种适用于企业级大数据分析的 Python 工具。在数据可视化、机器学习、深度学习等多方面都有涉及。不仅可以做数据分析，甚至可以用在大数据和人工智能领域，在数据分析和建模中不失为一种利器。

我们通过 Spyder 进行代码编写，因为它综合开发工具高级编辑，性能分析，调试和分析功能与数据探索，交互式执行，深度检查以及科学软件包的美观可视化功能相结合。大大提高我们对留言分类的数据分析效率。

2.3 总体流程图



3. 数据预处理

3.1 数据筛选

将题目所给的数据（C 题-附件.xlsx）excel 格式的样本数据导入 Anaconda 中以方便进行各种数据分析操作。成功导入数据至 Anaconda，开始对数据进行浏览和检查。

步骤 1：以分类效果为依据，删除冗余属性，选择出训练集中属性的优化组合。

步骤 2：使用分词工具对训练集的特征词进行分词。

步骤 3：对所得表进行特征词统计。

3.2 数据清理

对数据进行重新审查和校验，目的在于删除重复信息、纠正存在的错误，并提供数据一致性。清除重复样本，清除疑似错误异常的样本，清除偏离样本整体分布的样本。

在数据对象分为多个集合，在同一个集合里的对象有较高的相似度，而不同的集合之间的对象差别较大，我们采用**降维相关**算法处理数据集，对特征重要性进行排序，取重要性大的特征。

此外，将数据集划分为训练集和测试集了，我们利用一种常见的方法是将数据集按 80/20 进行划分，其中 80% 的数据用作训练，20% 的数据用作测试。

4. 数据优化及建模

我们使用 pandas、jieba、re、sklearn 进行数据优化及建模。pandas 是基于 NumPy 的一种工具，该工具是为了解决数据分析任务而创建的。还有 Pandas 是 python 的一个数据分析包，专注于 Python 数据包开发的 PyData 开发 team 继续开发和维护，属于 PyData 项目的一部分。

此外，pandas 纳入了大量库和一些标准的数据模型，提供了高效地操作大型数据集所需的工具。pandas 提供了大量能使我们快速便捷地处理数据的函数和方法。由于 pandas 是一种快速，强大，灵活且易于使用的开源数据分析和处理工具，对我们此次的数据优化有很大的帮助。

对于 jieba 分词，jieba 分词算法使用了基于前缀词典实现高效的词图扫描，生成句子中汉字所有可能生成词情况所构成的有向无环图(DAG)，再采用了动态规划查找最大概率路径，找出基于词频的最大切分组合，对于未登录词，采用了基于汉字成词能力的 HMM 模型，使用了 Viterbi

算法。它支持三种分词模式：精确模式：试图将句子最精确的切开，适合文本分析。全模式：把句子中所有可以成词的词语都扫描出来，速度非常快，但是不能解决歧义。搜索引擎模式：在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。

第三是 re，正则表达式本身是一种小型的、高度专业化的编程语言，而在 python 中，通过内嵌集成 re 模块，程序员可以直接调用来实现正则匹配。我们利用 re.sub 使用函数对匹配项的替换进行复杂的处理。

第四是 sklearn，sklearn 是 Scikit learn 的简称，是机器学习领域当中最知名的 python 模块之一。它有六种机械学习方式分别是 Classification 分类、Regression 回归、Clustering 非监督分类、Dimensionality reduction 数据降维、Model Selection 模型选择、Preprocessing 数据预处理。而我们用到 sklearn 的机器学习的方式有:Classification 分类，Preprocessing 数据预处理。

对于这次的建模我们用到两个模型，一个是高斯朴素贝叶斯模型，另一个是 OvR 模型。高斯朴素贝叶斯算法是一种特殊类型的 NB 算法，它特别用于当特征去油连续值时，同时假定所有特征都遵循高斯分布，即正态分布。

我们选择使用高斯朴素贝叶斯，将数据集拆分为训练集和测试集。OvR 原理：n 种类型的样本进行分类时，分别取一种样本作为一类，将剩余的所有类型的样本看做另一类，这样就形成了 n 个二分类问题，使用逻辑回归算法对 n 个数据集训练出 n 个模型，将待预测的样本传入这 n 个模型中，所得概率最高的那个模型对应的样本类型即认为是该预测样本的类型。

5. 算法（程序）的改进与推广

5.1 算法的改进

高斯朴素贝叶斯算法认为所有条件属性对决策分类的重要性一样，即权重都是 1，这样会让许多与分类无关的属性与其他关键属性具有同样的权重值，因此，只是简单统计词频是不合理的。基于此，众多科研人员将各种文本赋值算法应用于朴素贝叶斯分类器的构造中。

文献【1】提出了赋予不同属性不同权值的加权朴素贝叶斯 WNB（weighted naive bayes）模型，文献【2】提出 APNBC（adjusted probability bayes classifier）模型则通过二次加权调整后验概率后进行分类。

TFIDF 是用来评估一个关键词对语料库或文件集中某份文件的重要程度的常用算法，特征词在留言中的 TF-IDF 值越大，则表示该词对该留言的贡献值越大，反之，则该词对留言影响越小。但是将传统 TFIDF 算法应用到朴素贝叶斯分类器有一个缺点，没有考虑到类间、类内词汇的分布情况，即某词汇若在某留言中高频率出现，则该词汇的分类性能显然很强。

但是，根据经典 TFIDF 算法，如某词汇在很多留言中高频率出现，则其 IDF 较小，反之，IDF 较大。这同样会降低 NB 算法的分类性能。这对分类器决策性能有较大的影响，所以我们采用文献【3】中提出的改进 TFIDF 算法。

5.2 算法的推广

我们用到相似度计算里的欧氏距离和余弦相似度。欧氏距离是最常用的距离计算公式，衡量的是多维空间中各个点之间的绝对距离，当数据很稠密并且连续时，这是一种很好的计算方式。因为计算是基于各维度特征的绝对数值，所以欧氏度量需要保证各维度指标在相同的刻度级

别。

余弦相似度用向量空间中两个向量夹角的余弦值作为衡量两个个体间差异的大小。相比距离度量，余弦相似度更加注重两个向量在方向上的差异，而非距离或长度上。

另外，我们还用到聚类分析里的 k 均值聚类算法和层次聚类算法。k 均值聚类是对所有聚类的变量有个要求就是必须是连续性数值变量既然是连续性数值变量,那么就可以就求出 n 个变量的均值点,然后将每个个案的变量均值与总体的均值点在空间中比较,就可以找出位置上接近的点作为一类。

而层次聚类是聚类算法的一种，通过计算不同类别数据点间的相似度来创建一棵有层次的嵌套聚类树。在聚类树中，不同类别的原始数据点是树的最低层，树的顶层是一个聚类的根节点。这些算法都可以被推广使用。

6. 结论

经过一系列的工作，我们基本完成了应用文本挖掘提升政府的管理水平和施政效率这一目标。事实证明，政府部门通过应用大数据可以大幅度提升生产力和工作效能，并有效降低管理成本。

在本文所涉及的算法中，一个不足之处就是过程较为复杂，进行了一些可优化的重复性的操作。由于时间较为紧张，我们期待会有更多的时间对算法进行优化，提高结果的正确率与算法的效率。

我们认为还有很多值得改进的地方，一个良好的模型应该是基于简单的过程与精炼的算法就能得到不错的效果，因此努力的工作之一就是简化过程，精简算法。这一步还有很大的提升空间，据我们调查，该部分可通过采集标签集，通过聚类算法对采集的数据进行训练，相信如此

将会得到更好的效果。

以大数据助推政务管理精准化，面对越来越复杂多变的社会事务，政府应借助大数据手段进行政务管理，准确把握、及时发现问题，从而进一步提升政府监督管理的精准性和针对性。同时，通过深度数据挖掘分析，实时响应、处理公共事件和公众诉求。

7.参考文献

【1】Webb G I,Pazzan M J. Adjusted probability Naive Bayesian induction[C]//Proceedings of the 11th Australian Joint Conference on Artificial Intelligence.1998:285-295.

【2】Zhang H,Sheng S L. Learning weighted Naive Bayes with accurate ranking[C]//Fourth IEEE International Conference on DataMining. 2004:567-570.

【3】张玉芳，彭时名，吕佳.基于文本分类 TFIDF 方法的改进与应用[J].数据采集与处理，2006，32（19）：46-49.

【4】朴素贝叶斯算法的改进与应用-孟令军 SonSunLove|2016-03-27

附录 1

```
##import matploatlib.pyplot as plt
##from wordcloud import WordCloud
##import data
def bisai(file = &apos;C:\\Users\\14712\\Desktop\\数学建模\\附件
2.xlsx&apos;):
    data = pd.read_excel(&apos;C:\\Users\\14712\\Desktop\\数学
建模\\附件 2.xlsx&apos;, header=None)
    data.columns = [&apos;lybh&apos;,&apos;lyyh&apos;,&apos;zt
&apos;,&apos;sj&apos;,&apos;lly&apos;,&apos;fl&apos;]
    ly = data.lly
    label = data.fl
    n = 33
    ##ly 与 label 的合并 留言与分类
    shuju=pd.merge(label,ly, left_index=True,right_index=True,how=&ap
os;left&apos;)
     ##取相同的数量
    a=shuju[shuju[&apos;fl&apos;]==&apos;城乡建设&apos;].sample(n)
    b=shuju[shuju[&apos;fl&apos;]==&apos;环境保护
&apos;].sample(n)
    c=shuju[shuju[&apos;fl&apos;]==&apos;交通运输
&apos;].sample(n)
    d=shuju[shuju[&apos;fl&apos;]==&apos;教育文体
&apos;].sample(n)
    e=shuju[shuju[&apos;fl&apos;]==&apos;劳动和社会保障
&apos;].sample(n)
```

f=shuju[shuju['fl']=='卫生计生'].sample(n)

g=shuju[shuju['fl']=='商贸旅游 '].sample(n)

data_new=pd.concat([a,b,c,d,e,f,g],axis=0)

shuju['fl'].value_counts()

data_new.shape

daly=data_new.lly

##对留言进行预处理和分词操作

dapro=daly.apply(lambda x: re.sub('[\t|\n|\u3000|\xa0|\r]', '',x))

daly_cut=dapro.apply(jieba.lcut)

##去停用词

stop_word = pd.read_csv(r'C:\Users\14712\Desktop\数学建模 \stopword.txt', encoding ='GB18030', sep='hahaha ', header=None)

##stop_word = ['≮', '≯ ', '≠', '≮', ' ', '会 ', '月', '日 ', '–'] + list(stop_word.iloc[:, 0])

daly_after_stop=daly_cut.apply(lambda x:[i for i in x if i not in list(stop_word.iloc[:,0])])

word_count = collections.Counter(adaly)

word_count = word_count.most_common(7000)

word_list=[x[0] for x in word_count]

labels =data_new.loc[daly_after_stop.index,'fl']

labels.value_counts()

adaly = daly_after_stop.apply(lambda x: ' '.join(x

))

adaly.shape

return adaly, daly_after_stop, labels

bisai()

from bisai import bisai
##导入特征提取与分割 和高斯朴素贝叶斯模型
from sklearn.naive_bayes import GaussianNB
##from sklearn.ensemble import BaggingClassifier
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer, TfidfTransforme
r

from sklearn import preprocessing
from sklearn import datasets
from sklearn.metrics import accuracy_score
from sklearn.tree import DecisionTreeClassifier

adaly, daly_after_stop, labels = bisai()

tree = DecisionTreeClassifier(criterion='entropy', max_depth=N
one)

##分割成训练集和测试集；获取其特征个数 测试集对其进行维度的共享!! 使其的列数相同了
data_tr,data_te,labels_tr,labels_te = train_test_split(adaly, labels,random_state

```
=0, test_size=0.2)
countVectorizer = CountVectorizer()
data_tr = countVectorizer.fit_transform(data_tr)
data_te = CountVectorizer(vocabulary=countVectorizer.vocabulary_).fit_trans
form(data_te)

##获取训练集和测试集的  tf-idf 权值向量
X_tr = TfidfTransformer().fit_transform(data_tr.toarray()).toarray()
X_te = TfidfTransformer().fit_transform(data_te.toarray()).toarray()

model = GaussianNB()
model.fit(X_tr, labels_tr)
model.score(X_te, labels_te)
```