基于数据挖掘的智慧政务

摘要：随着网络时代越来越发达，“智慧政务”逐步成为群众和政府之间的一种新的沟通桥梁，但随着而来的是工作人员需要逐条整理分类留言，其工作量、工作难度是不可低估的，那么基于数据挖掘的方法对文本进行深度的分析，获取文本中的价值是解决以上难题的主要方法。

在本次数据挖掘过程中，我们首先利用 Jieba 搜索引擎模式对留言数据、答复数据进行分词、基于四种停用词库过滤停用词，用朴素贝叶斯和支持向量机构建文本分类器，选择具有 91%左右正确率的支持向量机分类器作为最终的一级标签分类模型；其次，通过一级标签分类模型为留言数据打上一级标签，并基于一级标签、地点将留言进行初步的划分后求取同一类别中相似度大于 0.6 的留言定义为同一个问题，将问题的时间跨度、赞同反对数、重复次数作为热度指标建立热度评判模型（见公式 2.4），模型结果见表 2.1 与表 2.2；最后，基于答复意见的相关性、完整性、时效性建立答复质量评判模型（见公式 3.3），模型结果见表 3.1。

综上，我们的挖掘结果能够有效地降低工作人员的工作量和工作难度，一级标签的分类准确率在可接受的范围内，也存在改进的空间，热度评判模型的时效性和准确性很高，答复评价模型指标选取不够全面，还有多种角度可以深入挖掘。

关键字：智慧政务，文本分词，分类器，热度评判，答复质量评判

1 挖掘目标

本次建模使用 Python 语言，针对自互联网公开来源的群众问政留言文本记录及政府相关部门对留言的答复文本数据，在对文本记录进行简单的预处理、分词、去除停用词后，建立朴素贝叶斯分类器、支持向量机分类器、热度评判准则、答复质量评判准则的数据挖掘模型，解决以下问题：（1）对留言数据进行一级标签分类（2）给出问题的热度评判标准，找出热点问题（3）给出政府相关部门答复的评判标准。

2 分析方法与过程

2.1 总体流程

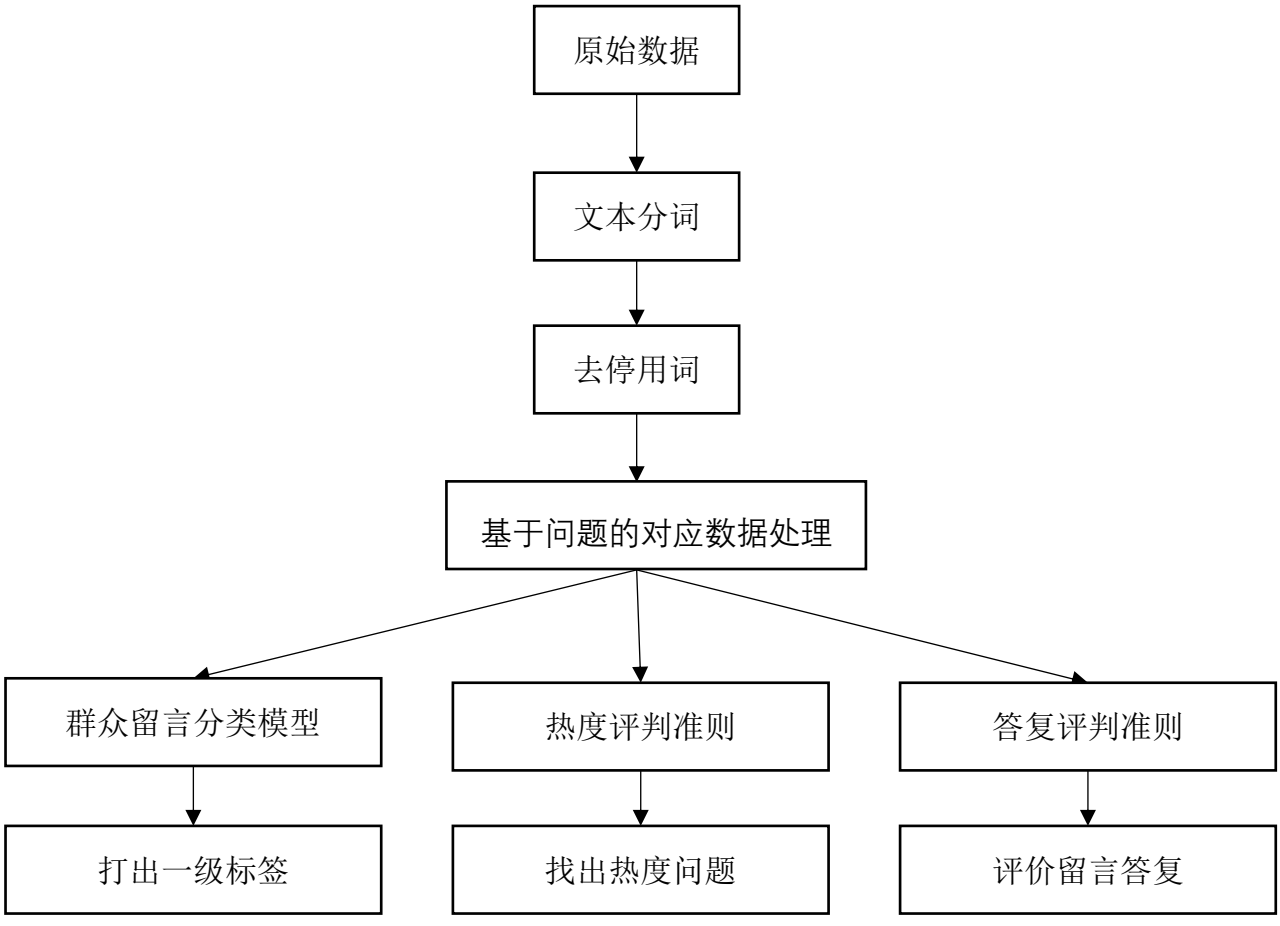

图 1.1 总体流程图

本论文的分析流程可大致分为以下步骤：

第一步：数据预处理，包括文本分词、去除停用词、打乱原始数据的顺序并按照 7:3 的比例划分为训练集和测试集；

第二步：通过机器学习算法对留言文本数据进行分析；

第三步：计算留言之间的相似度划分为不同类别的问题，基于时间、赞同数、反对数、重复次数建立热度评判准则为每一类问题打分；

第四步：针对答复的相关性、完整性、时效性建立答复质量评判准则。

2.2 具体步骤

2.2.1 文本数据预处理

首先，将群众留言分类对应到数据附件 2 可以发现，留言一级标签的确定主要取决于留言主题和留言详情。由此对于附件 2 的数据，需要对留言主题和留言详情进行处理。

其次，将热点问题挖掘对应到数据附件 3 可以发现，此问题要找到留言之间的相似程度，并给出留言热度的评分和留言中指定的特定地点或特定人群。对于附件 3 中包含的 4326 条留言数据，直接求每条留言之间的相似度是不现实的，所以我们首先通过群众留言分类的解决方法为每条留言打上一级标签，再按照留言中的地点划分为多个类别，最后计算被划分到一类的留言之间的相似度，再根据决定问题热度的指标建立热度评价模型。

最后，将答复意见评价对应到数据附件 4 可以发现，答复与留言的相关性可以考虑用两者之间的相似程度表示，答复的完整性根据文本特征来定义，答复的时效性通过时间间隔来权衡，当以上三个指标都能够准确表达时，再通过相关性、完整性、时效性指定评判答复的质量。

综上，对留言数据的预处理主要有：留言主题、留言详情、答复意见的中文分词、停用词过滤，基于一级标签、地点对留言进行初步的划分，同类留言数据的时间数据、点赞数、反对数、总条数的数据集成，答复与留言的时间间隔集成，答复意见的文本特征分析与集成；除此以外在训练模型时需要将原始数据划分为训练集合测试集，所以需要打乱原始数据的顺序，防止测试集出现了训练集中未出现过的一级标签。

2.2.1.1 留言文本分词

（1）分词方法的选择

分词是将连续的字序列按照一定的规范重新组合成词序列的过程。目前中文分词算法包括：基于字符串匹配的分词方法、基于理解的分词方法和基于统计的分词方法，我们考虑到建模采用 Python 语言，所以选择目前 Python 中最佳的中

文分词组件 jieba 进行分词。

（2）选择 Jieba 分词的原因

① 支持繁体分词，自带词库中的词数量超过 16 万，分词范围广

② 支持自定义词库，在实际操作中更方便

③ 当文本中出现新词时可以选择隐马尔可夫模型 HMM 进行分词，HMM 是一种基于概率的统计分析模型，能够有效地提高分词的准确性

④ 包含三种分词模式：精准模式、全模式、搜索引擎模式，不同的模式有不同的优劣，可以根据实际情况选择不同的模式。

（3）Jieba 分词的模式

全模式：将句子中所有可以相邻且可以成词的词语扫描出来，速度快，不能解决歧义，分词效果如图 1.2、图 1.3 和图 1.4：

```
# 全模式
seg_list = jieba.cut("这是“泰迪杯”数据挖掘挑战赛，“智慧政务”中的文本挖掘应用", cut_all=True)
print("【全模式】：" + "/ ".join(seg_list))
```

```
【全模式】：这/ 是/ “/ 泰/ 迪/ 杯/ ”/ 数据/ 数据挖掘/ 挖掘/ 挑战/ 挑战赛/ ，“/ 智慧/ 政务/ ”/ 中/ 的/ 文本/ 挖掘/ 应用
```

图 1.2 全模式分词效果图

精准模式：将句子精准地进行切分，分词效果如图：

```
# 精准模式
seg_list = jieba.cut("这是“泰迪杯”数据挖掘挑战赛，“智慧政务”中的文本挖掘应用", cut_all=False)
print("【精确模式】：" + "/ ".join(seg_list))
```

```
【精确模式】：这是/ “/ 泰迪杯/ ”/ 数据挖掘/ 挑战赛/ ，/ “/ 智慧/ 政务/ ”/ 中/ 的/ 文本/ 挖掘/ 应用
```

图 1.3 精准模式分词效果图

搜索引擎模式：在精准模式的基础上，将长词进行划分，效果如图：

```
# 搜索引擎模式
seg_list = jieba.cut_for_search("这是“泰迪杯”数据挖掘挑战赛，“智慧政务”中的文本挖掘应用")
print("【搜索引擎模式】：" + "/ ".join(seg_list))
```

```
【搜索引擎模式】：这是/ “/ 泰迪杯/ ”/ 数据/ 挖掘/ 数据挖掘/ 挑战/ 挑战赛/ ，/ “/ 智慧/ 政务/ ”/ 中/ 的/ 文本/ 挖掘/ 应用
```

图 1.4 搜索引擎模式分词效果图

仅仅通过个例进行测试不能判断出三种模式中选择哪一种模式更佳，由此后续将使用这三种模式进行分词，选择整体效果最佳者。

2.2.1.2 停用词过滤

停用词是指在信息检索中，为节省存储空间和提高搜索效率，在处理自然语言数据（或文本）之前或之后会自动过滤掉某些字或词，这些字或词即被称为 Stop Words（停用词），它们是我们日常生活中常用但意义不大的词语，去掉停用词后的词语序列能够表达的意思更明确，训练的模型会更佳准确。

停用词都是人工输入、非自动化生成的，生成后的停用词会形成一个停用词表，没有一个明确的停用词表能够适用于所有的工具。目前常用的停用词库有中文停用词库、哈工大停用词库、四川大学停用词库、百度停用词库，我们选择汇总四种停用词库并去掉重复的停用词作为新的停用词库进行过滤。

因此，将分词后的词语序列与停用词库进行匹配，若匹配成功则删去该词，通过对比过滤效果可以看出，停用词库中的词语并不全面，过滤的效果欠佳，所以我们去除了过滤停用词后词语序列中仅一个字的情况，如下例中：通过过滤停用词，删除了“的”、“应用”和标点符号，但“中”并没有去除，这样的特殊情况将被直接删除。

仅分词的效果如图 1.5：

这是，“，泰迪杯，”，数据，挖掘，数据挖掘，挑战，挑战赛，，，“，智慧，政务，”，中，的，文本，挖掘，应用

图 1.5 仅分词的结果图

分词并过滤停用词的效果如图 1.6：

['这是', '泰迪杯', '数据', '挖掘', '数据挖掘', '挑战', '挑战赛', '智慧', '政务', '中', '文本', '挖掘']

图 1.6 分词后过滤停用词的结果图

2.2.1.3 基于一级标签、地点对留言进行初步的划分

为了减少求取留言之间相似度的时间复杂度和空间复杂度，我们认为应该先划分具有一定相似性的留言后，再计算相似留言之间的相似度。那么划分相似度的标准是什么？

（1） 同一级标签划分

对于描述同一事件的留言，它们所属的一级标签也必须是统一的一级标签，所以在群众留言分类问题（问题一）中，通过分类模型能够找出某一留言所属某一一级标签的最大概率，那么我们就将留言所属的最大概率的一级预测标签作为留言的一级标签。

如图 1.6 为一级标签为卫生计生的留言，留言以下标表示。

卫生计生:[16, 81, 160, 232, 333, 355, 447, 571, 602, 694, 742, 835, 846, 908, 936, 1002, 1113, 1127, 1400, 1518, 1558, 1631, 1637, 1638, 1646, 1811, 1871, 2006, 2175, 2244, 2408, 2447, 2493, 2614, 2661, 2685, 2706, 2764, 2768, 3103, 3104, 3162, 3181, 3198, 3295, 3439, 3443, 3577, 3597, 3647, 3752, 3925, 3932, 3987, 3994, 3995, 4088, 4201, 4216, 4219, 4226, 4228, 4233, 4234, 4236, 4302, 4304]

图 1.7 一级标签为卫生计生的留言预览图

（2） 同一地点划分

通过分析留言的构成，它们多表达为对某一确定地点发生的事件的描述和建议，但地点的表述参差不齐，有的提到了完整的 X 市 X 区（县）X 地，有的提到了 X 区 X 地，而有的留言只提到了 X 地，所以想要找到每条留言之间相同的地点是一个繁琐的过程。

地点划分的步骤：

第一步：以“市”作为关键字匹配留言主题，匹配成功划分到市 X 类，匹配失败时，以“市”为关键字匹配留言详情，匹配成功划分到市 X 类，匹配失败划分到其他市类，并对相同市级的地点进行汇总；

如图 1.8 为某一一级标签下按照第一步划分的结果，留言用下标表示：

```
'I市': [673, 673, 673, 673],
'L市': [952, 1272, 1519, 1519, 1519, 4278, 4303],
'K市': [1111, 2695],
'C市': [1180, 1908, 2966],
'H市': [1519, 1519, 1519, 1519, 1519, 1519, 1519, 1519, 1519, 4278],
'G市': [1866, 1866],
'B市': [2201, 4135, 4135],
'F市': [2372, 3176, 3654],
'M市': [3044, 3176, 3176, 3176, 3176, 3654, 3654, 3654, 3654]},
```

图 1.8 一级标签为卫生计生的留言预览图

第二步：以“区”和“县”为关键字匹配市 X 类和其他市类中的留言，匹配成功划分为 X 市 X 区类或 X 市 X 县类（由于其他市类留言出现了对区县的表述，若匹配到的区县存在与市 X 类中出现的区县相同，我们就为这类留言补全市级），市 X 类中匹配失败的留言划分到市其他类。

如图 1.9 为某市类下按照第二步匹配到的区、县和市其他类的情况：

```
dict_keys(['A3区', '其他', 'A4区', 'A2区', 'A7县', 'A8县', 'A
6区', 'A5区', 'A1区', 'L5县', 'A区', 'L6县', 'K4县', 'K1区',
'K3县', 'F6县', 'B4区', 'F区', 'K9县', 'C区', 'M14县', 'F5
县', 'M11县', 'B区', 'E6县'])
```

图 1.9 基于地点划分的留言预览图（部分）

综上划分的结果可以表示为公式（1.1）：

$$
\begin{cases}
\text{城乡建设：}\begin{cases}
\text{市类：}\begin{cases}
\text{市 A 类：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\cdots\text{：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\text{市其他类}
\end{cases}\\
\text{其他市类}
\end{cases}\\
\cdots\text{：}\begin{cases}
\text{市类：}\begin{cases}
\text{市 A 类：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\cdots\text{：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\text{市其他类}
\end{cases}\\
\text{其他市类}
\end{cases}\\
\text{卫生计生：}\begin{cases}
\text{市类：}\begin{cases}
\text{市 A 类：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\cdots\text{：}\begin{cases}\text{市 A 区 B}\\ \cdots\\ \text{市 A 县 C}\end{cases}\\
\text{市其他类}
\end{cases}\\
\text{其他市类}
\end{cases}
\end{cases}
\tag{1.1}
$$

2.2.1.4 问答数据处理

对于问答数据的处理主要包括：留言详情与答复意见的分词，停用词过滤、留言与答复的时间间隔集成、答复意见的文本特征分析与集成。

（1） 留言详情与答复意见的分词

对于答复的相关性，我们将其定义为答复意见与留言详情中能够精确匹配的关键词的个数越多，相关性越强。分别对留言详情、答复意见进行分词处理，采用的分词方法、停用词过滤方法同 2.2.1.1，此外相似程度以关键词的匹配情况来定义，所以此处分词的模式选择精准模式，分词的效果较好，分词的结果也不会繁多。

（2）答复意见的文本特征分析与集成

对于答复的完整性，我们将其定义为答复文字中文本特征越强，答复的完整性越强。通过查阅资料，能够描述答复性句子的文本特征有：一、句子的长度，它包括题目的长度和描述的长度，一般高质量问题的长度较长；二、标点符号，

句子中不应该没有标点符号，但也不能有太多的标点符号。经过对答复意见的观察发现，答复之间长度的差异是非常大的，我们认为句子的长度在对答复完整性的评判上能够起到很大的作用，由此将句子的长度作为衡量答复完整性的指标，长度越长完整性越大。因此分别计算答复意见的句子长度。

（3）留言与答复的时间间隔集成

对于答复的时效性，我们将其定义为答复与留言的时间间隔越短，时效性越强。通过对时间数据的处理转换时间数据的格式，并进行计算获得留言与答复的时间间隔。

2.2.2 群众留言分类

群众留言数据量十分庞大、范围也极广，对留言进行初步的分类再分派到各个相应的职能部门进行处理既能减少工作人员的工作量还能使得群众的问题得到及时的解决。用机器学习中的算法构造分类器进行文分类是目前常用的方法，分类器的构造方法很多，如贝叶斯方法、决策树方法、基于实例的学习方法、人工神经网络方法、支持向量机方法、基于遗传算法的方法、基于粗糙集的方法、基于模糊集的方法等等，不同的算法有不同的优劣、适用场合，同时 Python 提供了一个强大的自然语言工具包 NLTK，支持使用 Python 语言构建机器学习算法的分类器进行文本分类，通过初步筛选，我们决定分别使用朴素贝叶斯分类器和支持向量机分类器对文本进行分类。

2.2.2.1 朴素贝叶斯分类器原理

朴素贝叶斯分类器是一系列以假设特征之间强（朴素）独立下运用贝叶斯定理为基础的简单概率分类器。朴素贝叶斯分类是一种十分简单的分类算法，拥有朴素的思想基础：基于先验概率，对给出的待分类项，求解在此项出现的条件下各个类别出现的概率，哪个最大，就认为此待分类项属于哪个类别。虽然该算法的思想朴素，但根据历年来不同文章中用朴素贝叶斯分类器进行文本分类的实验可以发现在很多复杂的情形中它能够获得相当好的效果。

朴素贝叶斯分类器的分类过程分为两步，第一步：用包含标签的原始数据进行训练，构建分类器；即使用训练数据拟合模型。第二步：用构建好的分类器对未知标签的数据进行分类；即使用测试数据进行模型准确性检验。

2.2.2.2 朴素贝叶斯分类器分类结果

由于原始数据包括了留言主题、留言详情两种，Jieba 分词的模式包括了全模式、精准模式、搜索引擎模式三种，所以以下通过不同组合方式进行分类，选择准确率最佳的组合作为最终的群众留言分类方法。列举不同的组合如下：

A.留言主题作为原始数据、Jieba 分词模式为全模式进行分类

B.留言主题作为原始数据、Jieba 分词模式为精准模式进行分类

C.留言主题作为原始数据、Jieba 分词模式为搜索引擎模式进行分类

D.留言详情作为原始数据、Jieba 分词模式为全模式进行分类

E.留言详情作为原始数据、Jieba 分词模式为精准模式进行分类

F.留言详情作为原始数据、Jieba 分词模式为搜索引擎模式进行分类

不同组合下的分类准确率如下表 1.1：

表 1.1 不同组合下，贝叶斯分类器的正确率

组合	组合 A	组合 B	组合 C	组合 D	组合 E	组合 F
准确率	0.8331	0.8154	0.8374	0.8259	0.8407	0.8418

经过以上结果对比分析后得出结论，通过朴素贝叶斯构建的分类器，6 种组合之间的差距不大，最好的组合是将留言详情作为原始数据，选择搜索引擎模式的分词方式。获得此结果的原因可能是：部分留言主题并不能完全概括留言详情的内容，会出现一定的偏差；搜索引擎模式分词既能解决分词的歧义，还能保证较为全面的捕获句子的意思。

2.2.2.3 支持向量机分类器原理

支持向量机（SVM）算法被认为是文本分类中效果较为优秀的一种方法，它是一种建立在统计学习理论基础上的机器学习方法。该算法基于结构风险最小化原理，将数据集合压缩到支持向量集合，学习得到分类决策函数。

对于像文本分类这样的非线性分类问题，可以通过非线性变换将它转化为某个维特征空间中的线性分类问题，在高维特征空间中学习线性支持向量机。由于在线性支持向量机学习的对偶问题里，目标函数和分类决策函数都只涉及实例和实例之间的内积，所以不需要显式地指定非线性变换，而是用核函数替换当中的内积。其执行过程同理朴素贝叶斯分类器。

2.2.2.4 支持向量机分类器分类结果

同理 2.2.2.2 节中 6 种不同的组合进行文本分类，分类准确率结果如表 1.2：

表 1.2 不同组合下，支持向量机分类器的正确率

组合	组合 A	组合 B	组合 C	组合 D	组合 E	组合 F
准确率	0.8461	0.8359	0.8549	0.8921	0.8947	0.9073

经过以上结果对比分析后得出结论，通过支持向量机构建的分类器，6 种组合之间有明显的差异，且对比贝叶斯分类器和支持向量机分类器的结果，后者的正确率都比前者高，最好的组合是将留言详情作为原始数据，选择搜索引擎模式的分词方式，获得此结果的原因同理贝叶斯分类器。

2.2.3 热点问题挖掘

对群众留言进行不同标签的大概划分虽然能够解决工作人员一定的工作量，但是对于留言这样庞大复杂的文字记录来说，很难做到及时给到群众满意的答复，所以从留言文本数据中挖掘出热点问题，并提取出热点问题涉及到的地点、人群等能够使相关部门更加有针对性地及时地解决一些重要的问题。

热点问题顾名思义就是一些民生关注度很大，被多次提及和赞同的问题。对于同一个问题，不同的人有不同的表述，那么就需要通过留言详情中的信息，求取不同留言之间的相似程度，相似度大于某一阈值我们就称之为同一个问题。如果直接计算每条留言之间的相似度，不仅时间复杂度是极大的，相似度阈值也无法准确的选择，因此我们计算相似度前将数据进行了初步的划分：①通过群众留言分类问题为留言打上一级标签；②对每一类一级标签中的数据按照地点划分为不同的类别。最后制定问题的热度评判规则，从而给出热点问题的排行情况。

2.2.3.1 相似度计算过程

通过数据处理对原始数据进行初步划分后，属于同一类别的留言分别计算两两之间的相似度，对于属于其他类（市其他类、其他市类）的留言，不能判断这样的地点的所属地，所以在计算相似度时需要放入每一种类别中计算。

文本相似度的计算过程包含文本分词、文本词向量生成、相似度计算三个步骤：

（1） 文本分词

在 2.1.1.3 节数据处理的过程中，需要为留言打上一级标签，打标签的同时已经对留言详情进行分词和去除停顿次，所以此处不再赘述。

（2） 文本词向量生成

词向量的生成我们通常计算词频-逆词频(TF-IDF)来用数值表示词语，那么句子中每个词的数值按照顺序排列组成词向量。

TF-IDF 是一种用来从文本文档中生成特征向量的简单方法。它为文档中的每个词计算两个统计值：一个是词频（TF），也就是每个词在文档中出现的次数，另一个是逆词频（IDF），用来衡量一个词在整个文档语料库中出现的频繁程度。这些值的乘积展示了一个词与特定文档的相关程度。

TF（词频）计算方式见公式 2.1，TF 越高，该词对文档来说越重要，但是 TF 不能作为文本相似度评价标准，对于一些常用的词语来说出现在一篇文档中的频率是非常高的，但同时出现在其他文档中也具有很高的词频，由此引入了 IDF（逆词频），即包含该词语的文档越少，IDF 就越大，计算方式见公式 2.2.最后通过 TF 和 IDF 计算结果相乘获得 TF-IDF 的得分见公式 2.3，一个词语的 TF-IDF 值越大，该词语在文档中的就越重要。

$$TF_t = \frac{N_t}{N} \tag{2.1}$$

$$IDF_t = \log\left(\frac{w}{W_t+1}\right) \tag{2.2}$$

$$TF - IDF_t = TF_t * IDF_t \tag{2.3}$$

其中，N_t表示文件中词 t 的频率，N 表示文件中词总数，W 表示文件总数，W_t表示包含词 t 的文件总数，那么留言的 TF-IDF 值就表示词在所属留言中的重要程度。

（3） 相似度计算

相似度的计算方法有最常见的欧氏距离、余弦相似度、切比雪夫距离等等，但对于文本数据之间的相似度应该考虑到语义之间的关系，所以我们用到了 Python 中的 Gensim 库，Gensim 是一个用于从文档中自动提取语义主题的 Python 库，它支持包括 TF-IDF，LSA，LDA 和 word2vec 在内的多种主题模型算法，支持流式训练，并提供了诸如相似度计算，信息检索等一些常用任务的 API 接口，是一个十分符合我们需求的工具。计算出两两留言之间的相似度后，通过查询相关资料、多次调试，我们最终选择将相似度阈值设置为 0.6。

在经过以上步骤后得到了一系列相似的留言，其中同一种问题超过 2 次被提到的问题种类有 367 种，如图 2.1 为划分到同一类别中的不同留言的描述，可以

发现我们获得的结果是可信的。

('A4区绿地海外滩小区距渝长厦高铁太近了',
'咨询A市绿地海外滩二期与长懿高铁问题',
'A4区绿地海外滩二期业主被噪音扰得快烦死了',
'A市至赣州高铁对绿地海外滩二期小区影响太大了',
'A4区绿地海外滩小区距长赣高铁最近只有30米不到，合理吗？',
'按照当前的高铁规划，A市绿地海外滩小区会饱受噪音困扰')

图 2.1 某一类别下留言的描述效果图

2.2.3.2 热度评价准则

对于问题热度的指标，我们认为涉及到三个方面，一、如果一个问题被多位群众上报，足以说明问题的重要性，二、留言的点赞数目越多，说明这个问题受到了群众大量的关注，反之，反对数降低了留言的可信度，三、如果一类问题被重复上报的时间间隔较短，说明这样的问题是比较紧急的，反之问题被上报的时间间隔较长，说明这样的问题没有得到解决或没有得到很好的解决，都是值得关注的。

由此，将某一问题被重复提及的次数作为问题的基础得分S_0，所获的点赞数作为问题的加分项S_1，所获的反对数作为问题的减分项S_2，通过最大最小归一化的方法将时间间隔线性转换到 0 到 1 之间（见公式 2.8），当时间间隔较小或较大时，问题的热度都属于较高的情况，因此时间间隔设置了一个权重，间隔越大或越小，权重越大，如图 2.2 所示。

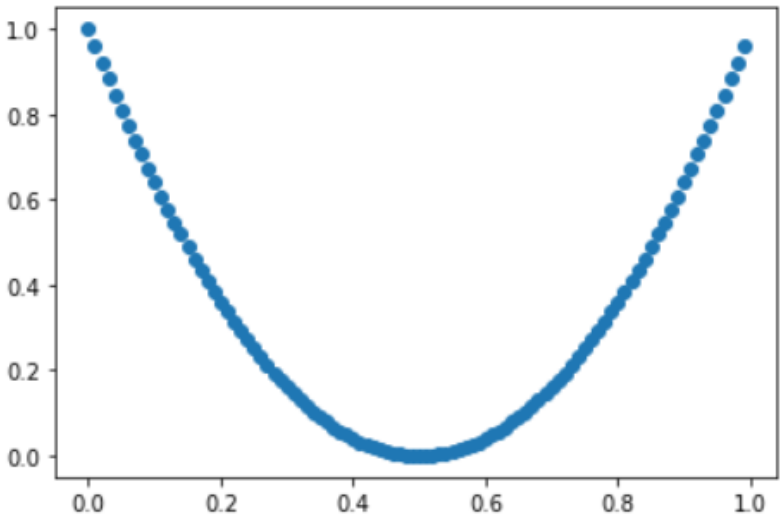

图 2.2 时间间隔权重走势图

那么每个问题的热度总得分S_hot，见如下公式：

$$S\_hot = (S_0 + S_1 + S_2) * （1 + W）$$ （2.4）

$$S_0 = n * 10$$ （2.5）

$$S_1 = 5 * \sum_1^n Support$$ （2.6）

$$S_2 = -5 * \sum_1^n Opposition$$ （2.7）

$$time' = \frac{time - time_{min}}{time_{max} - time_{min}}$$ （2.8）

$$W = 4 * (\text{time}' - 0.5) ** 2$$ （2.9）

其中 n 表示一类问题被提到的次数，Support 表示留言的赞同数，Opposition 表示留言的反对数，$time'$ 表示归一化后的时间间隔，time 表示原始时间间隔，$time_{min}$表示最初时间，$time_{max}$表示最晚时间。

2.2.3.3 热度问题结果展示

（1）问题热度得分展示，如图 2.3 给出部分类别的问题的得分情况，留言以下标表示：

```
([2962, 3887], 339.0),
([511, 1588], 310.0),
([924, 1671, 1965, 2607, 3008, 3045, 3411, 3638, 4055], 302.0),
([1345, 2029, 3274], 301.0),
([990, 2731, 3240], 299.0),
([3383, 3566], 298.0),
([2664, 3461], 295.0),
([827, 924, 1671, 1965, 3008, 3045, 3411, 3638, 4055], 263.0),
([3943], 260),
```

图 2.3 截取部分类别的热度得分情况

（3） 热点问题表预览

表 2.1 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	15697	2019/5/5 至 2019/9/19	A5 区五矿万境 K9 县	小区房子具有严重的质量问题，安全隐患大
2	2	11750	2019/2/21 至 2019/3/1	A 市 A4 区 58 车贷案	A 市 A4 区 58 车贷案
3	3	8795	2019/4/11	A 市金毛湾	配套入学的问题
4	4	6772	2019/8/23 至	A4 区绿地	小区距渝长厦高铁太近

5	5	1495	2019/9/5 2019/3/26 至 2019/4/9	海外滩小区 A6 区月亮岛	了，噪音扰人 沿线架设 110kv 高压电线杆的投诉

（4） 热点问题留言明细表预览

表 2.2 热点问题留言明细表预览（部分）

问题 ID	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
1	208636	A00077171	A 市 A5 区汇金路五矿万境 K9 县存在一系列问题	2019/8/19 11:34:04	我是 A 市 A5 区汇金路五矿万境 K9 县 24 栋的一名业主，我们小区一开始的定位是一个高端别墅小区...	0	2097
1	234086	A00099869	A 市五矿万境 K9 县房子的墙壁又开裂了	2019/6/20 9:30:44	五矿万境 K9 县的房子又出问题了，又是墙壁开裂！令人胆颤心惊！我是五矿万境 K9 县 36 栋的业主...	0	6
1	208069	A00094436	A5 区五矿万境 K9 县的开发商与施工方建房存在质量问题	2019/5/5 13:52:50	本人是 A5 区洞井街道汇金路五矿万境 K9 县 25 栋的业主，本人要投诉五矿地产及施工方…	0	2

1	215507	A000103230	A 市五矿万境 K9 县存在严重的消防安全隐患	2019/9/12 14:48:07	预交房 23 栋没有通往负一楼的楼梯，存在严重的消防安全隐患，开发商处理态度消极不予整改…	0	1
1	262599	A000100428	A 市五矿万境 K9 县房屋出现质量问题	2019/9/19 17:14:49	我是西地省 A 市五矿万境 K9 县的业主，2016 年所购买，于 2018 年 12 月 30 日交房。当初买房时…	0	0
1	275491	A00061339	A 市五矿万境 K9 县负一楼面积缩水	2019/9/10 9:10:22	关于五矿万境·K9 县负一楼面积缩水的反馈。我是有该楼盘的业主，在接到交房通知后经实地查验...	0	0
2	223297	A00087522	反映 A 市金毛湾配套入学的问题	2019/4/11 21:02:44	书记先生：您好！我是梅溪湖金毛湾的一名业主，和其他业主一样因为当初金毛湾的承若学校都是金毛建...	5	1762

2.2.4 热点问题挖掘

针对附件 4 中的答复意见数据进行分析发现，答复意见中存在的问题有：一、有的答复与留言完全不匹配，这对于政府部门来说是非常严重的问题，直接影响了群众对政府部门的信任度；二、有的答复语句不完整，根本不能构成一条通顺的语句，更谈不上与留言之间的匹配度；三、有的答复与留言的时间间隔太长，

对于一些紧急的问题根本不能得到及时的解决。综上，我们选择用我们自己的方式定义留言的相关性、完整性、时效性，并通过这三个指标来决定答复意见的质量。

2.2.4.1 相关性定义

经过分词、过滤停用词后，留言详情词序列、答复意见词序列之间的相关性定义为：留言详情词序列中每一个词分别匹配答复意见的词序列，匹配成功的词语数量作为这条答复与留言的相关数量。通过最大最小归一法将相关数线性映射到 0 到 1 之间，作为答复意见的相关度，见公式（3.1）。

$$Relevancy' = \frac{Relevancy - Relevancy_{min}}{Relevancy_{max} - Relevancy_{min}} \quad (3.1)$$

其中，$Relevancy$表示答复留言的相关数量，$Relevancy'$表示相关度，$Relevancy_{min}$表示全局最小相关数量，$Relevancy_{max}$表示全局最大相关数量。

2.2.4.2 完整性定义

获得每条答复数据的句子长度(词序列中词语的个数)后，答复意见的完整性定义为：答复句子的长度越长完整性越好，全局最小长度的答复的完整性定义为 0，全局最大长度的答复的完整性定义为 1，答复的全局总长度/答复的数量表示答复的长度，其完整性定义为 0.5。将上文的三个分界值作为输入数据，并基于神经网络模型预测每条留言的完整性$completeness$。

神经网络模型建立过程及结果如下：

（1） 选择神经网络模型的原因

神经网络是一种应用类似于大脑神经突触联接的结构进行信息处理的数学模型，通过不断的基于模型误差的方式逆向更新神经网络上每个神经元之间的权值，直到到达最大迭代次数或满足规定的最小误差值。而我们的需求是通过句子的长度预测其完整性，且输入值和输出值已知，神经网络的工作机制和结果都是满足要求的。

（2） 网络模型的构建

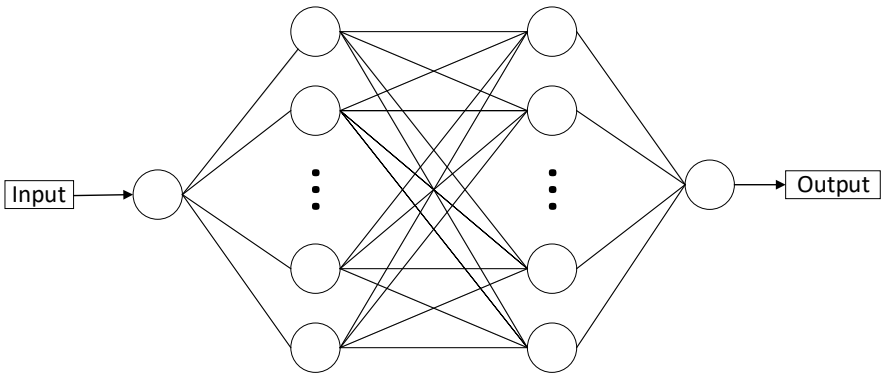

图 3.1 多层网络模型构成图

（3） 激活函数的选择

神经网络的激活函数包括：Sigmoid, ReLu, TanHyperbolic(tanh), softmax, softplus，我们选择 ReLu 作为激活函数。

相比于传统的神经网络激活函数，诸如逻辑函数（Logistic sigmoid）和 tanh 等双曲函数，ReLu 有着以下几方面的优势：

1.仿生物学原理：相关大脑方面的研究表明生物神经元的信息编码通常是比较分散及稀疏的。通常情况下，大脑中在同一时间大概只有 1%-4%的神经元处于活跃状态。使用线性修正以及正则化（regularization）可以对机器神经网络中神经元的活跃度（即输出为正值）进行调试；相比之下，逻辑函数在输入为 0 时达到 1/2，即已经是半饱和的稳定状态，不够符合实际生物学对模拟神经网络的期望。不过需要指出的是，一般情况下，在一个使用修正线性单元（即线性整流）的神经网络中大概有 50%的神经元处于激活态。

2.更加有效率的梯度下降以及反向传播：避免了梯度爆炸和梯度消失问题

3.简化计算过程：没有了其他复杂激活函数中诸如指数函数的影响；同时活跃度的分散性使得神经网络整体计算成本下降。

2.2.4.3 时效性定义

获得每个问答之间的时间间隔后，答复意见的时效性定义为：时间间隔越大，时效性越差，反之时间间隔越小，时效性越佳。通过最大最小归一法将时间间隔线性映射到 0 到 1 之间，作为答复意见的时效度，见公式（3.2）。

$$Eff\_time' = \left(1 - \frac{Eff_{time} - Eff_{time_{min}}}{Eff_{time_{max}} - Eff_{time_{min}}}\right) \quad (3.2)$$

其中，Eff_time 表示答复留言的时间间隔，Eff_time' 表示时效度，Eff_time_{min}表示全局最小相关数量，Eff_time_{max}表示全局最大相关数量。

2.2.4.4 答复评判准则

通过对答复意见的相关性、完整性、时效性的定义，三种指标由一个确定的数值表示，我们将评判答复质量得分$S_quality$制定为（见公式 3.3）：

$$S_{quality} = Relevancy' + Eff\_time' + completeness \quad (3.3)$$

2.2.4.4 答复评判结果展示

表 3.1 答复评判结果展示（部分）

质量排行	质量指数	留言编号	留言主题	留言时间	留言详情	答复意见	答复时间
1	2.5429975	9744	关于对 A 市迪和衡平中医医院在北京…	2016/10/25 12:47:15	我们是 A 市 A3 区北京园御小区的广大业主，现就 A 市迪和衡平中医医院在北京园御小区原售楼部违规建设医院…	一、关于“公示不符合国家卫生部明文规定”的问题…	2016/11/24 9:34:40
2	2.4171599	9199	A5 区教育局延恒大城公立幼儿园开办	2017/3/14 14:37:50	请求政府及时接收“恒大城”小区配套幼儿园…	1.恒大城幼儿园办园的相关依据…	2017/3/24 13:46:11
3	2.3138445	4331	质疑 A 市新城国际花都的装修价格	2018/10/24 10:55:47	经过查看在政府备案的该楼盘的装修价格的评估报告分析，我们业主对该楼盘的装修价格以及[政府发文]【2018】53 号文有诸多疑问与看法…	一、根据《营业税改征增值税试点有关事项的规定》…	2018/11/6 10:21:48
4	2.3047944	18413	反对在 A7 县江背镇乌川湖村兴建涵洞	2015/12/30	经调查，反映我镇镇长张登武不作为的信访人为黄尊富，为我镇乌川湖村楼梯坡组人，长期在外…	经镇综治办调查核实，信访人黄尊富，系我镇乌川湖村楼梯坡组村民…	2016/1/7

				2:26			10:5
				:45			9:47
			请求 A 市				
			政府及时	201		目前，学前教育以	201
	2.27		接收“恒大	7/3/	2017 年 2 月 26 日，在 A5 区政	“政府主导、社会	7/3/
5	535	9225	城”小区配	12	府、A5 区教育局、洞井街道、龙	参与、公办和民办	31
	3		套幼儿园	11:1	庭社区…	并举”为主要发展	16:3
			并开办公	6:24		原则…	3:12
			立园				

3 结论

本文通过对处理过的收集自互联网公开来源的群众问政的文本留言数据和政府部门的答复文本数据利用朴素贝叶斯、支持向量机、神经网络等方法建立了留言一级标签分类模型、热点问题评判模型、答复质量评判模型，对文本数据进行数据挖掘，实现了对文本留言的一级标签划分、基于群众对相关问题的反馈程度、认同程度以及问题存在的时间跨度进行了热度问题的搜索和评判、基于政府部门答复留言的相关性、完整性、时效性进行了答复质量的评判。以上结果有效地降低了工作人员的工作量和差错率，热点问题能够在较短的时间内捕获，紧急或民生关注度大的问题能够得到及时的解决，答复有了评判的标准后，群众收到的回复都是通过检验的意见，提高了民生对政府的满意度。

对于群众留言一级标签划分的问题，只用到了两种算法构建分类器，我们认为可以 3-4 种算法进行比较，比如基于神经网络、基于遗传算法建立分类器，或将算法进行组合建立分类器，以获得更高的准确率；对于热度问题的解决中，我们对留言进行了比较精细的划分再求留言之间的相似度，但是在词向量的生成上选择了 TF-IDF，其实还可以尝试更多的词向量计算方法进行结果的对比；对于答复质量的评判上，我们只考虑了答复的相关性、完整性、时效性，还可以加入更多的角度如：可解释性、专业性、礼貌性等进行更加深入的文本挖掘，以获得更加准确的答复评判标准。

参考文献

[1] 祁小军,兰海翔,卢涵宇,丁蕾锭,薛安琪.贝叶斯、KNN 和 SVM 算法在新闻文本分类中的对比研究[J]. 电脑知识与技术.2019(09)

[2] 曲凯扬.基于支持向量机的文本分类研究[J].无线互联科技.2016(05)

[3] 崔哲.基于朴素贝叶斯方法的文本分类研究[D].河北科技大学.2018

[4] 李丹.基于朴素贝叶斯方法的中文文本分类研究[D].河北大学.2011

[5] 邓海龙.Python 词向量训练与应用技术解析.语料库语言学.2019(09)

[6] zhbzz2007.结巴分词 1--结巴分词系统介绍,
https://www.cnblogs.com/zhbzz2007

[7] 迅速傅里叶变换. gensim 简介,
https://www.jianshu.com/p/e21b59a46e4c

[8] JH0lmes.分类器,
https://blog.csdn.net/JH0lmes/article/details/82790997

[9] 掌舵的鹰, 排序算法常用评价指标计算方式（AUC,MAP,NDCG,MRR）
https://blog.csdn.net/weixin_38405636/article/details/80675312