“智慧政务”智慧政务中的文本挖掘应用

摘要

近年来，随着网络问政平台逐步成为政府了解社情民意的重要渠道，各类社情民意相关的文本数据量不断攀升。因此，运用自然语言处理和文本挖掘的方法对来自互联网公开来源的群众问政留言记录以及相关部门对部分群众留言的答复意见的研究具有重大的意义。

问题 1 对网络问政平台的群众留言进行分类，采用 BERT 模型对文本分类。首先将数据随机选取 90%数据作训练集，10%作验证集，然后使用 BERT 中的预训练模型，将数据特征转换并训练模型，同时选取并保存验证集上得分最高的模型。最后加载训练阶段的最佳模型进行测试，并采用 F-Score 进行模型的评价。

问题 2 对热点问题挖掘，通过利用 excel 对留言时间进行排序。选取特定时间段的留言信息，利用 jieba 中文分词工具对留言详情信息进行分词，并通过 TF-IDF 算法得到每个留言主题的权重向量，采用 K-means 对权重向量进行聚类。然后统计点赞数和反对数这两列的信息，筛选出聚类后每类中群众集中反映的某些问题，将排名前五的问题也称为热点问题保存为热点问题表。同时将热点问题对应的留言信息保存为热点问题留言明细表。

问题 3 对答复意见的评价，利用计算留言详情和答复意见的文本相似度，计算留言时间和答复时间的时间差分数以及答复意见的完整性分数，这三个方面来评价分析答复意见的质量。

关键词：BERT，中文分词，K-means 聚类，word2vec，文本相似度

"Smart Government Affairs" Text Mining Application in Smart Government Affairs

Abstract

In recent years, as the online political inquiry platform has gradually become an important channel for the government to understand social conditions and public opinion, the amount of text data related to various social conditions and public opinion has been increasing. Therefore, the use of natural language processing and text mining methods is of great significance to the research on the records of the public's questioning messages from public sources on the Internet and the responses of relevant departments to some of the public's messages.

Question 1 categorizes the mass messages of the online questioning platform, and uses the BERT model to classify the text. First, randomly select 90% of the data as the training set and 10% as the verification set. Then use the pre-trained model in BERT to convert the data features and train the model. At the same time, select and save the model with the highest score on the verification set. Finally, load the best model in the training stage for testing, and use F-Score to evaluate the model.

Question 2 mines hot issues and sorts the message time by using excel. Select the message information of a specific time period, use jieba Chinese word segmentation tool to segment the message details, and obtain the weight vector of each message topic through the TF-IDF algorithm, and use K-means to cluster the weight vector. Then count the information of the likes and dislikes of the points, filter out some of the problems that are concentrated in the people in each category after clustering, and save the top five problems, also called hotspots, as a hotspot list. At the same time, the message information corresponding to the hotspot questions is saved as a list of hotspot questions.

Question 3 evaluates the reply opinion by calculating the details of the message and the text similarity of the reply opinion, calculating the time difference between the message time and reply time, and the completeness score of the reply opinion. These three aspects evaluate and analyze the quality of the reply opinion.

Keywords: BERT, Chinese word segmentation, K-means clustering, word2vec, text similarity

目　录

1. 挖掘目标

本次建模的目标是利用集自互联网公开来源的群众问政留言记录，及相关部门对部分群众留言的答复意见，利用 BERT[1]进行文本分类，利用 jieba 中文分词工具对留言详情信息进行分词，K-means 聚类的方法，文本相似度方法，达到以下三个目标：

（1）利用数据进行文本分类，分类结果以便后续将群众留言分派至相应的职能部门处理，同时解决了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。

（2）利用文本分词和文本聚类的方法对非结构化的数据进行文本挖掘，根据聚类结果，结合点赞数和反对数筛选热点问题，有助于相关部门进行有针对性地处理，提升服务效率。

（3）从多个角度分析相关部门对留言的答复意见，从而判断答复意见的质量。

2. 分析方法与过程

2.1 问题 1 的分析方法与过程

2.1.1 流程图

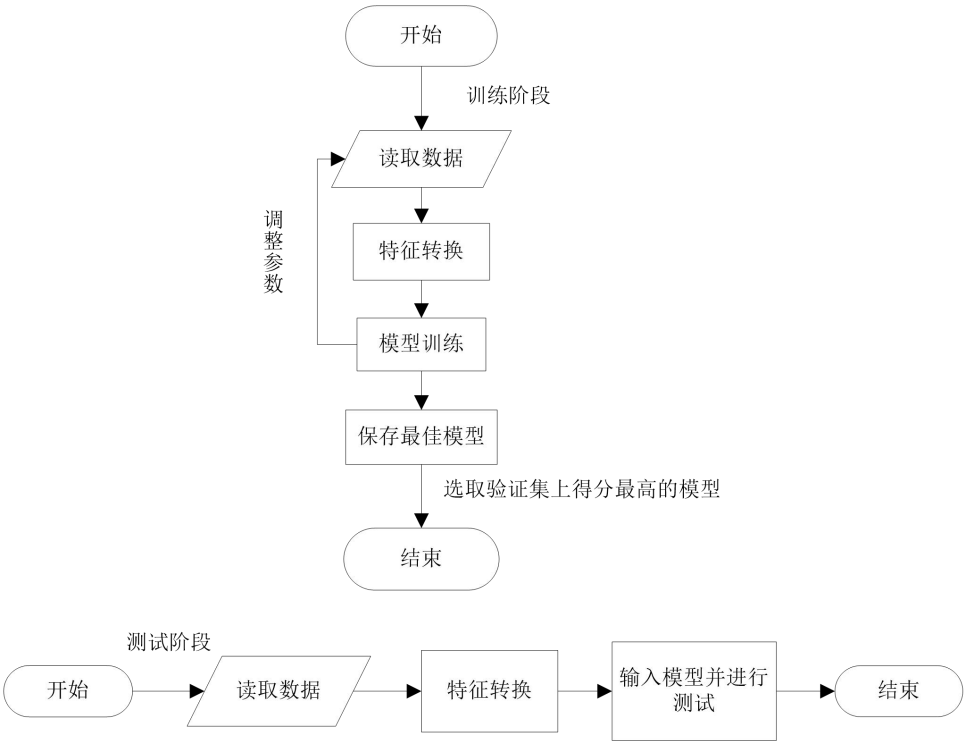


图 2-1 问题 1 流程图

2.1.2 读取数据

首先获得题目中给定的数据集，包括留言的编号，用户，主题，时间，详情和分类标签。将数据集中的数据随即打乱顺序，并按照 9：1 的比例将数据集划分为训练集和验证集。针对问题 1 群众留言分类，用数据读取模块中的两个方法分别读取数据集中留言详情和一级分类这两列的数据，一个方法是获取训练集的数据另一个方法是获取验证集的数据。

2.1.3 特征转换

特征转换就是将获取到的留言详情这列文本数据转换成特征，该特征由三部分组成：

（1） input_ids：分词后每个词语在 vocabulary 中的 id，补全符号对应的 id 为 0。

（2） input_mask：真实字符/补全字符标识符，真实文本的每个字对应 1，补全符号对应 0

（3） segment_ids：句子 A 和句子 B 分隔符，句子 A 对应的全为 0，句子 B 对应的全为 1。但是在多数文本分类情况下并不会用到句子 B。

转换完成后的特征值就可以作为输入，用于模型的训练和测试。

2.1.4 模型训练

（1） 预训练模型：BERT-Base, Chinese: L = 12, H = 768, A = 12

（2） 微调了批量大小、学习率和训练次数，分别是：

Batch size: 12

Learning rate (Adam):2e-5

Number of epochs: 3

（3） 当 train_steps 每训练 1000 次会在验证集上进行验证，并给出相应的精度值，如果准确值大于此前最高分则保存模型。如果 epoch 数超过先前设定的 num_train_epochs，则停止迭代。

2.1.5 模型测试

先加载模型，将测试数据测试读入到模型中，将测试结果打印在文件上。

对群众留言分类的评价指标是使用 F-Score 对分类结果进行评价：

$$F = \frac{1}{n}\sum_{n=i}^{n}\frac{2P_iR_i}{P_i + R_i} \qquad (2-1)$$

其中 P_i 为第 i 类的查准率，R_i为第 i 类的查全率。

2.2 问题 2 分析方法与过程

2.2.1 流程图

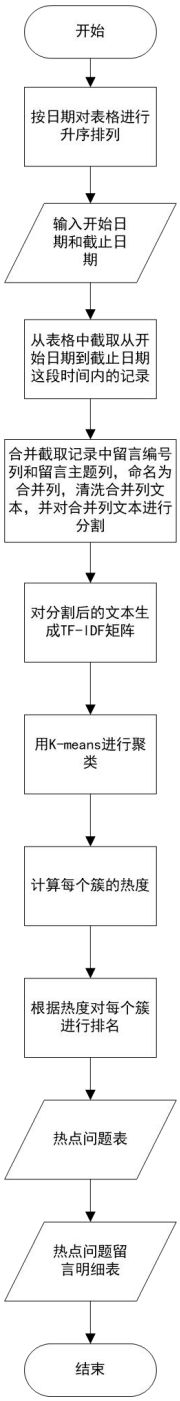


图 2-2 问题 2 流程图

2.2.2 数据预处理

2.2.2.1 留言主题信息的去空、去中文符号和英文符号

在数据分析中我们重点研究的是数据，但是不是每个数据都是我们需要分析的，这就需要我们去清洗数据，通过清洗数据，这样我们就能够保证数据分析出一个很好的结果，所以说一个干净的数据能够提高数据分析的效率，因此，数据清洗是一个很重要的工作，通过数据的清洗，就能够统一数据的格式，这样才能够减少数据分析中存在的众多问题，从而提高数据的分析的效率。在题目给出的数据中，留言主题信息中夹杂着很多空格，还有很多中文符号和英文符号。为了能让数据分析能有一个很好的结果，需要把这些空格、中文符号和英文符号去除掉。

2.2.2.2 对留言主题信息进行中文分词

在对留言主题信息进行分析挖掘之前，要先把非结构化的文本信息转换为计算机能够识别的结构化信息。在附件 3 中，以中文文本的方式给出了数据。为了便于转换，先要对这些留言主题信息进行中文分词。这里采用了 python 中的中文分词包 jieba 进行分词。jieba 的分词过程大体可分为三部：1.首先用正则表达式将中文段落粗略的分成一个个句子。2.将每个句子构造成有向无环图，之后寻找最佳切分方案。3.最后对于连续的单字，采用 HMM 模型将其再次划分。本步骤用的是 jieba 的精确分词模式，精确分词模式会将句子最精确地切开，非常适合做文本分析，使得能更好的实现中文分词效果。

2.2.2.3 TF-IDF 算法

在对留言主题信息分词后，需要把这些词语转换为向量，以供挖掘分析使用。这里采用[2]TF-IDF 算法，把留言主题信息转换为权重向量。TF-IDF 算法的具体原理如下：

第一步，计算词频，即 TF 权重（Term Frequency）。

$$\text{词频（TF）} = \text{某个词在文本中出现的次数} \tag{2-2}$$

考虑到文章有长短之分，为了便于不同文章的比较，进行“词频”标准化，除以文本的总词数或者除以该文本中出现次数最多的词的出现次数即：

$$\text{词频（TF）} = \frac{\text{某个词在文本中出现的次数}}{\text{文章的总词数}} \tag{2-3}$$

或

$$词频（TF）= \frac{某个词在文本中出现的次数}{该文本出现次数最多的词的出现次数} \quad (2\text{-}4)$$

第二步，计算 IDF 权重，即逆文档频率（Inverse Document Frequency），需要建立一个语料库（corpus），用来模拟语言的使用环境。IDF 越大，此特征在文本中的分布越集中，说明该分词在区分文本内容属性能力越强。

$$逆文档频率（IDF）= \log(\frac{语料库的文本总数}{包含该词的文本数+1}) \quad (2\text{-}5)$$

第三步，计算 TF-IDF 值（Term Frequency Document Frequency）。

$$TF\text{-}IDF = 词频（TF）\times 逆文档频率（IDF） \quad (2\text{-}6)$$

实际分析得出 TF-IDF 值与一个词在职位描述表中文本出现的次数成正比，某个词文本的重要性越高，TF-IDF 值越大。

2.2.2.4 生成 TF-IDF 向量

生成 TF-IDF 向量的具体步骤如下：

（1） 对留言主题描述提供的词，合并成一个集合，计算每条留言主题描述对于这个集合中词的词频，如果没有则记为 0；

（2） 生成每条留言主题描述的 TF-IDF 权重向量，计算公式如下：

TF-IDF=词频（TF）×逆文档频率（IDF） (2-7)

2.2.3 留言主题的分类

生成留言主题描述的 TF-IDF 权重向量后，根据每条留言主题的 TF-IDF 权重向量，对留言主题进行分类。这里采用 K-means 算法把职业类型分成 7 类。

K-means 聚类[3]的原理如下：

假设有一个包含 n 个 d 维数据点的数据集 $X = \{X_1, X_2, \ldots, X_i, \ldots, X_n\}$，其中$X_i \in R^d$，K-means 聚类将数据集 X 组织为 K 个划分 $C = \{C_k, i = 1,2\ldots,K\}$。每个划分代表一个类 ck，每个类 ck 有一个类别中心 μi。选取欧式距离作为相似性和距离判断准则，计算该类内个点到聚类中心 μi 的距离平方和

$$J(C_K) = \sum_{X_i \in C_K} ||X_i - \mu_K||^2 \quad (2\text{-}8)$$

聚类目标是使各类的距离平方和 $J(C_K) = \sum_{K=1}^{K} J(C_K)$最小，

$$J(C) = \sum_{K=1}^{K} J(C_K) = \sum_{K=1}^{K} \sum_{X_i \in C_i} ||X_i - \mu_i||^2 = \sum_{K=1}^{K} \sum_{i=1}^{n} d_{Ki} ||X_i - \mu_i||^2 \quad (2\text{-}9)$$

其中，$d_{Ki} = \begin{cases} 1, 若 X_i \in C_i \\ 0, 若 X_i \notin C_i \end{cases}$，所以根据最小二乘法和拉格朗日原理，聚类中心$\mu_K$

应该取为类别c_K类各数据点的平均值。

K-mean 聚类的算法步骤如下：

1、从 X 中随机取 K 个元素，作为 K 个簇的各自的中心。

2、分别计算剩下的元素到 K 个簇中心的相异度,将这些元素分别划归到相异度最低的簇。

3、根据聚类结果，重新计算 K 个簇各自的中心，计算方法是取簇中所有元素各自维度的算术平均数。

4、将 X 中全部元素按照新的中心重新聚类。

5、重复第 4 步，直到聚类结果不再变化。

6、将结果输出。

K-means 聚类[4]的算法流程图如下：

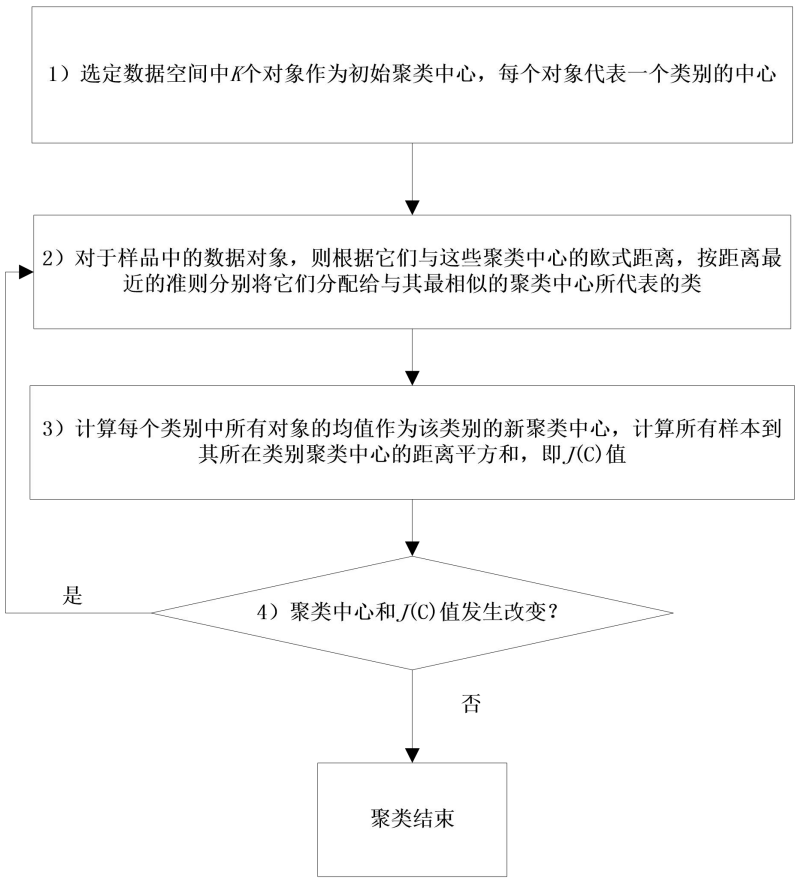


图 2-3 聚类算法流程图

利用按照开始时间和截止时间截取出来的样本进行分词、求 TF-IDF 向量，并利用 K-means 聚类，把样本分成 7 类，程序见 start_1.py、matrix_2.py 和 td_idf_3.py。

2.2.4 留言主题热度

经过上一个步骤的 K-means 聚类，已经得到了每个簇内留言主题相对应的记录，可以看到每条记录后边都有反对数和点赞数。因此我们可以采用 Reddit 评论排名算法计算每个簇的热度。代码见 hot.py。

2.2.4.1Reddit 评论排名算法

我们先做如下设定：

- 每个用户的投票都是独立事件。
- 用户只有两个选择，要么投好评，要么投差评。
- 如果投票总人数为 n，其中好评为 k，那么好评率 p 就等于 k/n。

根据统计学已经看出来了，p 服从一种统计分布，叫做“两项分布”(binomial distribution)。

p 越大，就代表这个项目的好评比例越高，越应该排在前面。但是，p 的可信性，取决于有多少人投票，如果样本太小，p 就不可信。由于 p 服从”两项分布”，因此我们可以计算出 p 的置信区间。所谓“置信区间”，就是说，以某个概率而言，p 会落在的那个区间。比如，某个产品的好评率是 80%，但是这个值不一定可信。根据统计学，我们只能说，有 95% 的把握可以断定，好评率在 75%到 85%之间，即置信区间是$[75\%，85\%]$。

通过上面的分析，我们就可以推断出，如果要给一个评论进行排名，就需要考虑一下内容：

- 计算每个评论的”好评率”
- 计算每个”好评率”的置信区间(以 95%的概率)。
- 根据置信区间的下限值，进行排名。这个值越大，排名就越高。

这样做的原理是，置信区间的宽窄与样本的数量有关。比如，A 有 8 张赞成票，2 张反对票；B 有 80 张赞成票，20 张反对票。这两个项目的赞成票比例都是 80%，但是 B 的置信区间(假定$[75\%，85\%]$)会比 A(假定$[70\%，90\%]$)窄得多，因此 B 的置信区间的下限值(75%)会比 A(70%)大，所以 B 应该排在 A 前面。

置信区间的实质，就是进行可信度的修正，弥补样本量过小的影响。如果样本多，就说明比较可信，不需要很大的修正，所以置信区间会比较窄，下限值会比较大；如果样本少，就说明不一定可信，必须进行较大的修正，所以置信区间会比较宽，下限值会比较小。

二项分布的置信区间有多种计算公式，最常见的是“正态区间”(Normal approximationinterval)，教科书里几乎都是这种方法。但是，它只适用于样本较多的情况(np >5 且 n (1-p) >5)，对于小样本，它的准确性很差。

1927 年，美国数学家 Edwin Bidwell Wilson 提出了一个修正公式，被称为“威尔逊区间”，很好地解决了小样本的准确性问题。Reddit 目前使用的是评论算法就是基于威尔逊得分区间 (Wilson score interval)[5]。具体的代码片段如下：

```
from math import sqrt

def _confidence(ups, downs):
    n = ups + downs

    if n == 0:
        return 0

    z = 1.0 #1.0 = 85%, 1.6 = 95%
    phat = float(ups) / n
    return (phat+z*z/(2*n)-z*sqrt((phat*(1-phat)+z*z/(4*n))/n))/(1+z*z/n)


def confidence(ups, downs):
    if ups + downs == 0:
        return 0
    else:
        return _confidence(ups, downs)
```

使用到的威尔逊得分区间具体公式如下：

$$\frac{\hat{p}+\frac{1}{2n}z_{1-\alpha/2}^2 \pm z_{1-\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}+\frac{z_{1-\alpha/2}^2}{4n^2}}}{1+\frac{1}{n}z_{1-\alpha/2}^2} \tag{2-10}$$

其中

- $\hat{p}$ 是好评率
- n 是总投票数
- Z(1-α/2)表示对应某个置信水平的 z 统计量，这是一个常数，可以通过查表得到。一般情况下，在 95%的置信水平下，z 统计量的值为 1.96。

可以公式看到，当 n 的值足够大时，这个下限值会趋向$\hat{p}$。如果 n 非常小(投票人很少)，这个下限值会大大小于$\hat{p}$。

实际上，起到了降低”好评率”的作用，使得该评论的得分变小、排名下降。

威尔逊得分区并不关心一个评论的投票数，而关心好评数和投票总数或采样大小的相对关系！

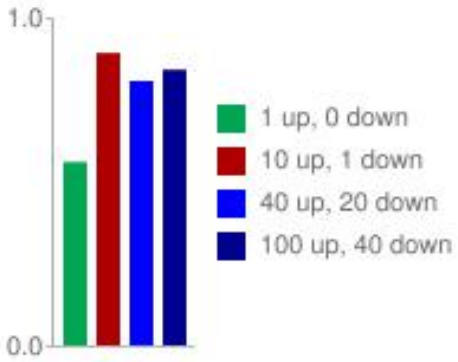


图 2-4 根据威尔逊得分区计算出来的值

上图是根据威尔逊得分区计算出来的值：一个评论有 1 个好评，没有差评，它的支持率是 100%，但是由于数据量过小，系统还是会把它放到底部。 但如果，它有 10 个好评，1 个差评，系统可能会有足够的信息把他放到一个有着 40 个好评，20 个差评的评论之前。因为我们基本确认当它有了 40 个好评的时候，它收到的差评会少于 20 个。较好的一点是，一旦这个算法出错了(算法有 15%的失效概率)，它会很快拿到更多的数据，因为它被排到了前面。

2.3 问题 3 分析方法与过程

针对问题三，决定对答复意见的相关性、完整性、及时性、可解释性这 4 个角度对答复意见的质量给出一套评价方案。

2.3.1 流程图

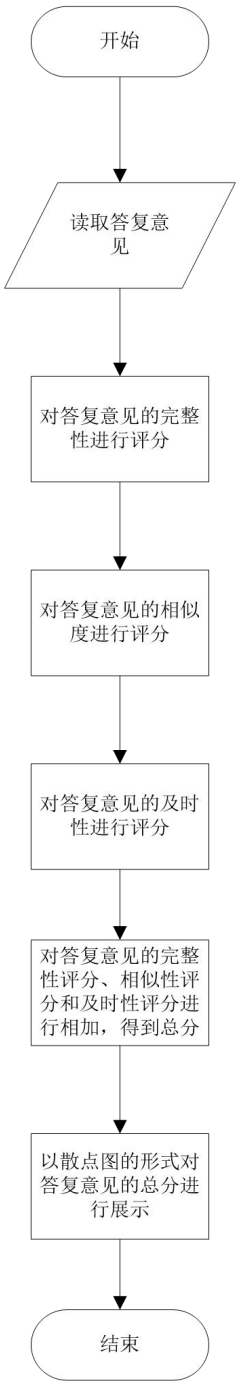


图 2-5 问题 3 流程图

2.3.2 答复意见的完整性评分

答复意见的格式规范性和答复意见的内容充分度决定着答复意见的完整性。答复意见的完整性评分由 8 个方面的评分组成。初始分数是 0 分。具体的评分规则如下：

（1） 答复意见的字数：答复意见的字数小于或等于 225，加 5 分；答复意见的字数大于 225 且小于或等于 450，加 8 分；答复意见的字数大于 450，加 11 分；

（2） 答复意见中出现了“你好”或“您好”字样，加 1 分。

（3） 答复意见中出现了“收悉”或“获悉”字样，加 1 分。

（4） 答复意见中出现了“回复如下”或“答复如下”字样，加 1 分。

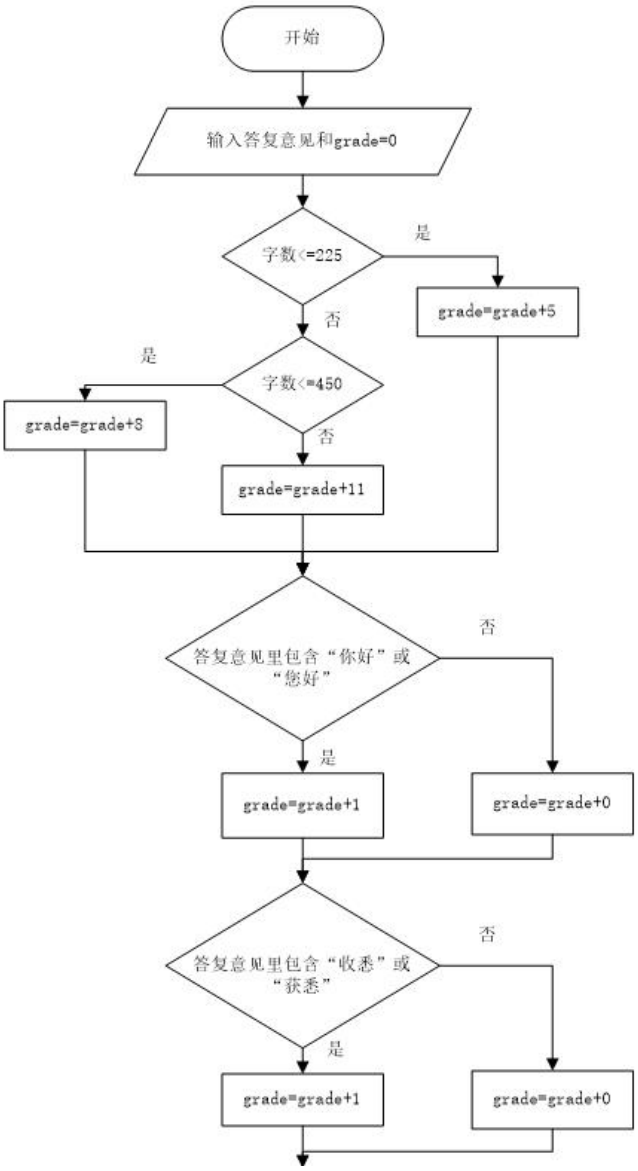
（5） 答复意见中出现了“一、”和“二、”字样，加 1 分。

（6） 答复意见中出现了“1、”和“2、”字样，加 1 分。

（7） 答复意见中出现了“特此回复”字样，加 1 分。

（8） 答复意见中出现了“年”和“月”和“日”字样，加 1 分。

答复意见的完整性评分流程如图 2-6 所示：



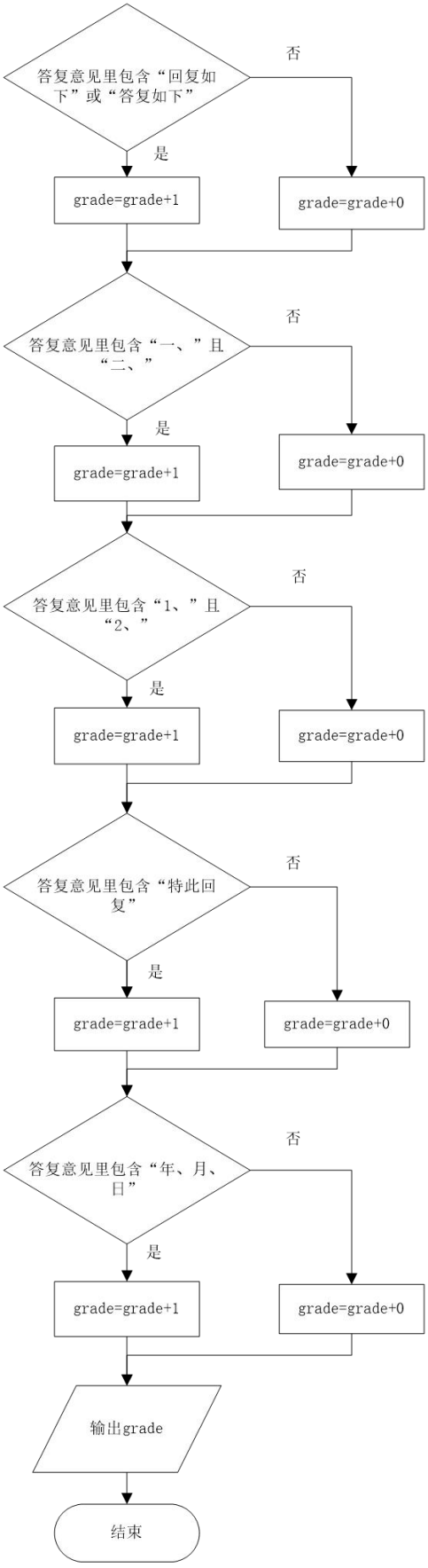


图 2-6 答复意见的完整性评分流程图

2.3.3 答复意见的相似度评分

对答复意见的相似度进行评分主要是考察答复意见和留言详情的相似度，答复意见的相似度评分体现出了答复意见和留言详情的吻合程度。具体的评分规则如下：

（1） 对留言详情和答复意见进行分词。

（2） 利用步骤（1）分好的词得到句子的词频向量。

（3） 用余弦相似度计算留言详情和答复意见的相似度。

（4） 利用步骤（3）得到的相似度得出答复意见的相似度评分。

答复意见的相似度评分流程如图 2-7 所示：

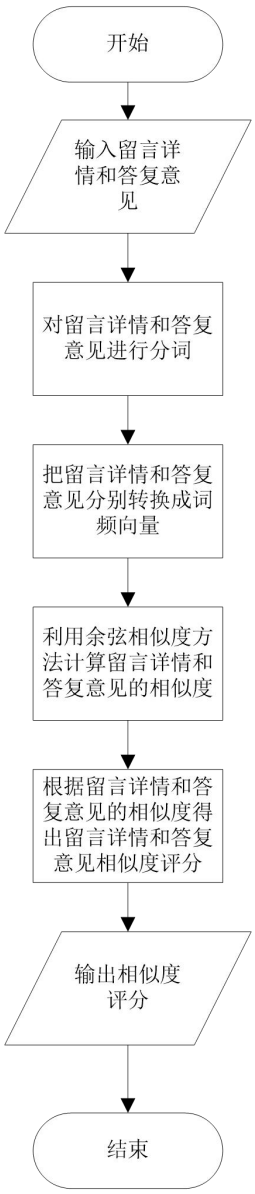


图 2-7 相似度评分流程图

2.3.3.1 余弦相似度

余弦相似度[6]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。余弦值越接近 1，就表明夹角越接近 0 度，也就是两个向量越相似，这就叫"余弦相似性"。

我们知道，对于两个向量，如果他们之间的夹角越小，那么我们认为这两个向量是越相似的。余弦相似性就是利用了这个理论思想。它通过计算两个向量的夹角的余弦值来衡量向量之间的相似度值。余弦相似性公式如下：

$$\cos(\theta) = \frac{\sum_{i=1}^{n}(X_i \times y_i)}{\sqrt{\sum_{i=1}^{n}(X_i)^2} \times \sqrt{\sum_{i=1}^{n}(y_i)^2}} \tag{2-11}$$

$$= \frac{a \cdot b}{||a|| \times ||b||}$$

计算两个句子向量
句子 A：(1，1，2，1，1，1，0，0，0)
和句子 B：(1，1，1，0，1，1，1，1，1)的向量余弦值来确定两个句子的相似度。
计算过程如下：

$$\cos(\theta)$$
$$= \frac{1 \times 1 + 1 \times 1 + 2 \times 1 + 1 \times 0 + 1 \times 1 + 1 \times 1 + 0 \times 1 + 0 \times 1 + 0 \times 1}{\sqrt{1^2 + 1^2 + 2^2 + 1^2 + 1^2 + 1^2 + 0^2 + 0^2 + 0^2} + \sqrt{1^2 + 1^2 + 1^2 + 0^2 + 1^2 + 1^2 + 1^2 + 1^2 + 1^2}}$$
$$= \frac{6}{\sqrt{7} \times \sqrt{8}}$$
$$= 0.81$$

2.3.4 答复意见的及时性评分

由于留言反映的是群众当时面对的问题，如果过了很长的时间再去回复群众的留言，可能群众等到回复的时候，当时的问题已经得到解决了。所有答复意见具有一定的时效性。针对答复意见的时效性，决定对答复意见的及时性进行评分。留言时间减去答复时间得到时间差。对附件 4 所有记录的时间差进行求和，然后取平均值，得到平均值为 17 天。具体的评分规则如下：初始分数为 5 分，时间差每高于 17 天 1 天，在 5 分的基础上减掉 0.25 分，最终分数小于 0 分的按 0 分计入；时间差每低于 17 天 1 天的，在 5 分的基础上加上 0.25 分。

答复意见的及时性评分流程如图 2-8 所示：

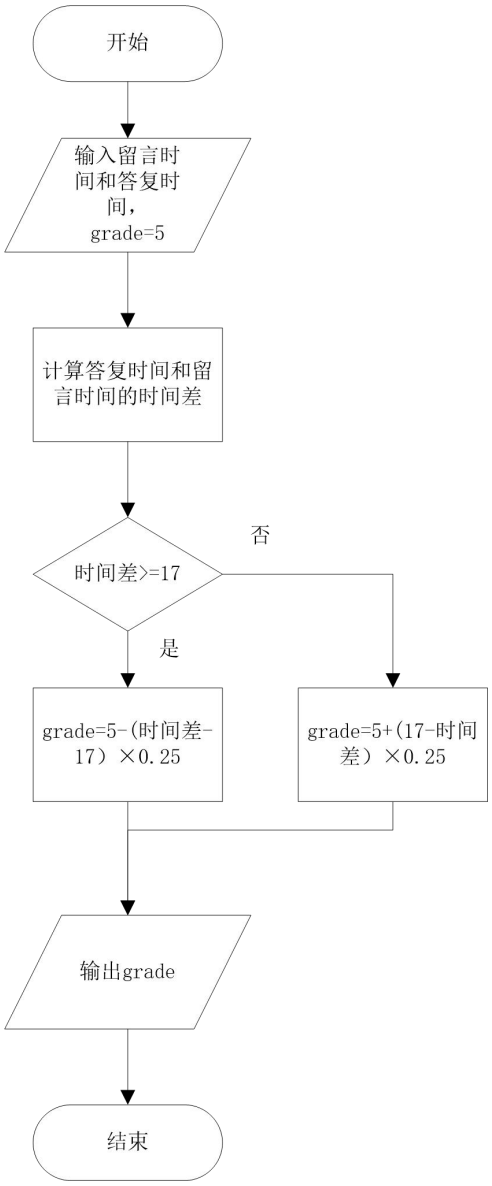


图 2-8 答复意见的及时性评分流程图

2.3.5 可解释性

最后的成绩以散点图的形式展现出来，更加清晰明了。

3. 结果分析

3.1 问题 1 结果分析

问题 1 对群众留言分类，在训练阶段，我们随机选取所有数据中的 90%进行训练模型，10%进行验证模型保存准确率最高的模型为 91.86%，在测试阶段，测试出来的结果采用公式（2-1）计算出 F-Score 进行模型评估，模型基本满足

技术需求。

3.2 问题 2 结果分析

在问题 2 中，采用了 2019 年 1 月 1 日至 2019 年 3 月 12 日这个时间段的数据记录。

热点问题表如表 3-1 所示：

表 3-1 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	0.999475555650579	2019/1/11 21:12:34 至 2019/3/1 22:12:30	58 车贷受害者	58 车贷诈骗案没有处理妥当
2	2	0.982183960543755	2019/1/2 20:27:07 至 2019/2/11 14:09:40	关注城市土地规划的市民	关于城市土地规划的相关问题
3	3	0.977802953522311	2019/1/7 15:00:59 至 2019/3/11 11:33:16	诈骗案件的举报者	关于诈骗案件的举报
4	4	0.963918213333197	2019/1/3 22:05:24 至 2019/1/22 23:32:29	被拖欠工资的工人	工人迟迟拿不到工资
5	5	0.963918213333197	2019/2/13 10:22:45 至 2019/3/5 8:19:16	南山十里天池	关于南山十里天池存在的问题

热点问题留言明细表如表 3-2 所示：

表 3-2 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	223787	A0003486	全国典型诈	2019/1/11 21:12	4.任由犯罪	0	0
1	272413	A0001060	息出，A4区	2019/1/14 20:23:57	但我们出借	2	0
1	254532	A0001060	案近半年没	2019/1/14 22:08:20	58官网上	3	0
1	272858	A0006178	件为什么不	2019/1/16 23:21:21	立案近半年	0	0
1	264119	A0008444	A4区公安	2019/1/19 9:47:23	员，未查封	0	0
1	214238	A0006178	对58车货一	2019/1/20 22:28:40	想问一下：	2	1
1	218132	A0001060	A市58车贷	2019/1/29 19:15:49	办案警官毛	0	0
1	268251	A0001060	年毫无进展	2019/2/2 15:03:05	管和资产，	25	0
1	240554	A0002916	跑路美国，	2019/2/10 20:58:40	步嫌保护伞	6	0
1	220711	A0003168	注A市A4区	2019/2/21 18:45:14	消息总是失	821	0
1	217032	A0005654	贷特大集资	2019/2/25 9:58:37	苏纳弟弟	790	0
1	194343	A0001061	贷案警官应	2019/3/1 22:12	并没有跟进	733	0
2	256358	A0008032	西泉塘昌和	2019/1/2 20:27	原厂房和闲	29	0
2	233542	A0008032	西泉塘昌和	2019/1/2 20:27:26	原厂房和闲	24	0
2	239670	A0008032	西泉塘昌和	2019/1/11 15:46:04	业有限公司	41	0
2	273391	A0001095	公岭路以西	2019/1/15 9:37:38	的山体的	3	0
2	204713	A0008566	塘公交首末	2019/1/22 21:23:48	塘公交首末	4	0

3.3 问题 3 结果分析

根据设计的评价方案得出的每条答复意见的分数如表 3-3 所示：

表 3-3 答复意见的分数

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	及时性评分	相似度评分	完整性评分	总分
2549	A0004558	A2区景蓉	2019/4/2	2019年4月	现将网友	2019/5/1	5.5	5.68	12	23.18
2554	A0002358	A3区潇楚	2019/4/2	潇楚南路	网友“A0	2019/5/9	5.75	5.21	11	21.96
2555	A0003161	请加快提	2019/4/2	地处省会	市民同志	2019/5/9	5.75	5.88	12	23.63
2557	A0001107	在A市买公	2019/4/2	尊敬的书	网友“A0	2019/5/9	5.75	4.2	13	22.95
2574	A0009233	关于A市公	2019/4/2	建议将“	网友“A0	2019/5/9	5.5	6.2	4	15.7
2759	A0007753	A3区含浦	########	欢迎领导	网友“A0	2019/5/9	1.5	3.79	3	8.29
2849	A0001008	A3区教师	2019/3/2	尊敬的胡	网友“A0	2019/5/9	0	5.49	3	8.49
3681	UU00812	反映A5区	2018/12/	我做为一	网友“UU	2019/1/29	2.25	5.2	16	23.45
3683	UU008792	反映A市美	2018/12/	我是美麓	网友“UU	2019/1/16	5.25	4.39	12	21.64
3684	UU008687	反映A市洋	2018/12/	胡书记好	网友“UU	2019/1/16	5.25	6.31	3	14.56
3685	UU008220	反映A2区	2018/12/	我家住在	网友“UU	2019/3/11	0	4.12	12	16.12
3692	UU008829	A5区鄱阳	2018/12/	胡书记：	网友“UU	2019/1/29	1.75	5.31	12	19.06
3700	UU00877	A4区万国	2018/12/	尊敬的书	网友“UU	2019/1/14	5.25	4.84	4	14.09
3704	UU008148	举报A市芒	2018/12/	尊敬的领	网友“UU	2019/1/3	8	4.49	4	16.49
3713	UU008122	建议增开	2018/12/	建议增开	网友“UU	2019/1/14	5	4.13	4	13.13

答复意见分数的分布图如图 3-1 所示：

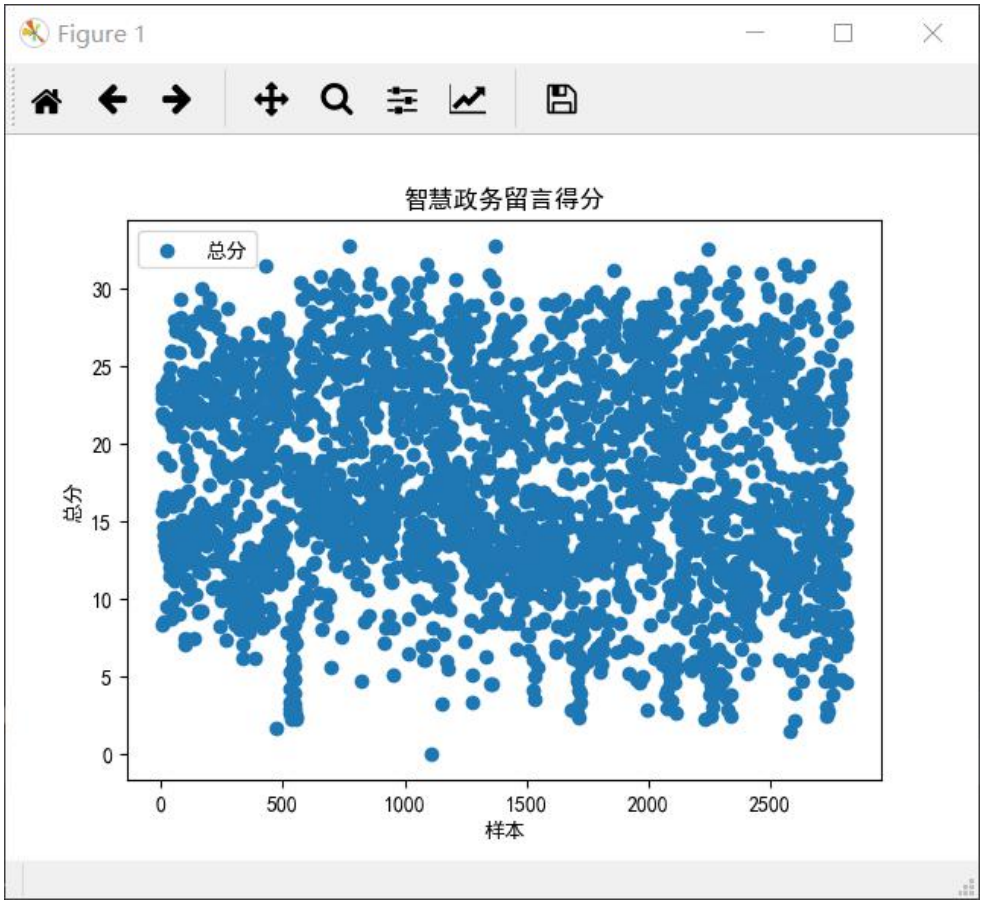


图 3-1 答复意见分数分布图

4. 结论

对来自互联网公开来源的群众问政留言记录，及相关部门对部分群众留言的答复意见进行分析研究，及时了解社情民意，提升政府的管理水平和施政效率有极大的推动作用。传统的文本解读已经不能满足数据量庞大的各类社情民意相关的文本信息。而本文利用自然语言处理和文本挖掘的方法解决存在的问题。

由实验结果可看出，在处理网络问政平台的群众留言分类时，利用深度学习中的模型去处理分类问题效率高而且差错率低。采用根据 K-means 聚类方法，统计某时刻群众留言问题类型，并且筛选出热点问题，有助于相关部门进行有针对性地处理，提升服务效率。采用文本相似度算法等算法从多个角度进行分析答复意见的质量，从分析结果可以看出，相关政府针对民众的问题留言能够及时，准确，全面的回复。

5. 参考文献

[1] Devlin J,Chang M W,Lee K,et al.BERT:Pre-training of deep bidirectional transformers for language understanding[EB/OL].(2018-10-11)[2020-01-17].https://arxiv.org/abs/1810.04805

[2] 施聪莺,徐朝军,杨晓江.TFIDF 算法研究综述[J].计算机应用,2009,29(S1):167-170+180.

[3] 杨俊闯,赵超.K-Means 聚类算法研究综述[J].计算机工程与应用,2019,55(23):7-14+63.

[4] 侯泽民,何建仓.K-Means 聚类算法在大学生智慧就业平台中的应用研究[J].福建电脑,2018,34(12):18+37

[5] 徐林龙,付剑生,蒋春恒,林文斌.一种基于威尔逊区间的商品好评率排名算法[J].计算机技术与发展,2015,25(05):168-171.

[6] 王春柳,杨永辉,邓霏,赖辉源.文本相似度计算方法研究综述[J].情报科学,2019,37(03):158-168.