C 题

摘要

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。因此，运用自然语言处理和文本挖掘的方法来研究“智慧政务”中的信息具有重大意义。

对于问题一，提取附件 2 的留言主题和留言详情并合并。首先进行数据预处理：将留言主题与留言详情合并对非结构化数据进行去重、去敏、去除无意义符号；之后利用正则表达式获取重要词汇将其增加到个人词库，接着进行 Jieba 分词并加载停用此表，去除停用词；然后基于 TF-IDF 权重策略，将文本中的词语转换为词频矩阵，并统计每个词语的 TF-IDF 值；最后使用支持向量机（Support Vector Machine, SVM）多分类算法对留言内容进行一级标签分类。通过 F1-Score 对分类模型完成评价。

对于问题二，首先进行前期数据处理：第一，将留言主题与留言详情合并作为留言内容，进行正则化，Jieba 分词，去除停用词等，第二，利用 Hanlp 包进行命名实体识别，第三，将命名实体识别以及留言主题的分词结果加入到留言内容的分词结果中，以增加其权重，至此完成数据处理部分；然后使用 BIRCH（Balanced Iterative Reducing and Clustering using Hierarchies）聚类算法对所有留言问题进行聚类；最后对每一类留言问题，计算其热度指数，找出热度排名前五的问题。

对于问题三，本小组提出了一个合适的解决方案，答复意见的质量评价包括：答复内容本身的质量评价（包括答复的完整性）、问题和答复的匹配度（包括答复的相关性、可解释性）计算。首先构建正例词典和反例词典，问题中出现的词汇进入问题的正反例词典，答复中出现的词汇进入答复的正反例词典。本小组提出一种基于主题聚类的方法进行问题和答复的匹配度计算，然后采用最大熵统计模型作为各个特征的融合框架，以实现对单个问答对质量的评价，最终给每个问答对有一个正确的评价分值。

关键词：中文分词、TF-IDF 权重策略、SVM 多分类算法、BIRCH 聚类算法、基于主题的聚类算法、最大熵统计模型。

Abstract

In recent years, with the online questioning platforms such as WeChat, Weibo, mayor's mailbox, and sunshine hotline, etc., it has gradually become an important channel for the government to understand public opinion, gather public wisdom, and gather popular sentiment. The work of the relevant departments, which mainly rely on manpower for message division and hotspot sorting, has brought great challenges. Therefore, it is of great significance to use natural language processing and text mining methods to study information in "smart government affairs".

For question one, extract the message subject and message details of Annex 2 and merge them. First perform data preprocessing: combine the subject of the message with the details of the message to deduplicate, desensitize, and remove meaningless symbols; then use regular expressions to obtain important vocabulary and add it to the personal vocabulary, then Jieba word segmentation Load and deactivate this table to remove stop words; then convert the words in the text into a word frequency matrix based on the TF-IDF weight strategy, and count the TF-IDF value of each word; finally use Support Vector Machine (Support Vector Machine) , SVM) multi-classification algorithm for first-level tag classification of message content. Complete the evaluation of the classification model through F1-Score.

For question two, first perform the preliminary data processing: first, combine the subject of the message with the details of the message as the content of the message, perform regularization, Jieba word segmentation, remove stop words, etc., second, use the Hanlp package for named entity recognition, third , Add the name entity recognition and the word segmentation result of the message subject to the word segmentation result of the message content to increase its weight, and then complete the data processing part; then use BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) clustering algorithm Perform clustering. Finally, for each category of message questions, calculate the heat index to find the top five questions.

For question three, the group proposed a suitable solution. The quality evaluation of the response comments includes: the quality evaluation of the response content (including the completeness of the response), the matching degree of the question and the response

(including the relevance and interpretability of the response) Sex) calculation. First, construct a positive example dictionary and a negative example dictionary, the words appearing in the question enter the positive and negative example dictionary of the question, and the words appearing in the answer enter the positive and negative example dictionary of the answer. This group proposes a method based on topic clustering to calculate the matching degree of questions and answers, and then uses the maximum entropy statistical model as the fusion framework of each feature to achieve the evaluation of the quality of a single question and answer. A correct evaluation score.

Keywords: Chinese word segmentation, TF-IDF weighting strategy, SVM multi-classification algorithm, BIRCH clustering algorithm, topic-based clustering algorithm, maximum entropy statistical mode3l

目 录

1.挖掘目标

本次建模目标是利用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 jieba 中文分词工具对群众留言进行分词，SVM 分类模型及 PCA 降维及 BIRCH 聚类算法，达到以下三个目标：

1）利用文本分词和文本分类的方法对非结构化的数据进行文本挖掘，根据分类结果和 F1-Score 得分，建立关于留言内容的一级标签分类模型。

2）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。帮助相关部门及时发现热点问题，进行有针对性地处理，提升服务效率。

3）对答复意见本身的质量以及问题与答复的匹配度分别进行评价，并建立合适的多特征融合框架以实现对每个问答对的质量进行评价，给有关部门进行实时答复反馈，有助于他们进行实时调整。

2.分析方法与过程

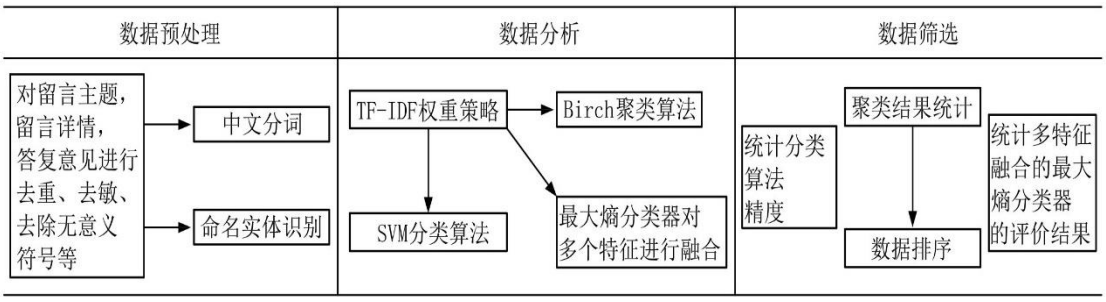

图 2.1 总流程图

2.1 问题一分析方法与过程

2.1.1 流程图

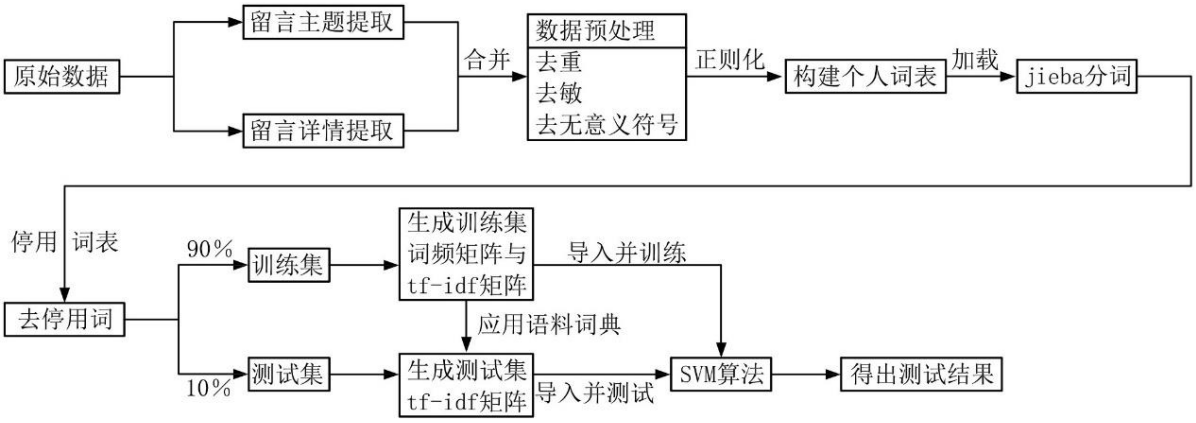

图 2.2 问题一流程图

2.1.2 数据预处理

（1）合并文本并去重、去敏、去无意义符号

由于留言主题和留言详情都存在可作为分类依据的关键词，且这些关键词可能不重复，为提高模型精度，将两者作合并处理。之后的操作都只针对合并后的数据。

在进行数据的抽取或者导入时，可能会发生数据的重复，因此需要进行去重操作。同时，数据中存在的x序列及空格等对留言分类没有意义且会产生干扰，也应进行剔除。这里主要使用了 python 中的 apply()函数。

（2）创建个人词库

利用正则表达式，获取文本中与字母和数字相关的关键词，例如找出文本中的“A 市”、“B2 区”、“M 县”或者“2019 年”、“300 万”、“20 人”等信息。这些带有字母或者数字的词汇是一条文本中的关键词，需要挑选出来建立个人词库使得进行 Jieba 分词时不会将其分开。

（3）对群众留言进行中文分词

在对群众留言表进行数据挖掘分析之前，要把非结构化的文本数据转化为计算机能够识别的结构化数据。中文分词以词作为基本单元，使用计算机自动对中文文本进行词语的切分，这样方便计算机识别出各语句的重点内容。因此，为了便于转化，本文对群众留言表中所给出的留言主题进行中文分词操作。

在进行分词之前，根据给出的群众留言表增加了个人词库，以保证这些重要词汇不会被分词，使分词工具更加适用于该文本数据的分词操作。这里利用正则表达式提取留言内容中带有数字或字母的关键词。

采用的 python 中的中文分词包 Jieba 来进行中文分词。Jieba 分词属于概率语言模型分词，其基本原理为：

a.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成词情况所构成的有向无环图(DAG)。

b.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。

c.对于未登录词，采用基于汉字成词能力的 HMM 模型，使用 Viterbi 算法。

部分留言内容原句、Jieba 分词结果如下表所示：

表 2.1 jieba 分词结果

留言内容原句	Jieba 分词结果
A 市西湖建筑集团占道施工有安全隐患。A3 区大道西行便道，未管所路口至加油站路段，人行道包括路灯杆，被圈西湖建筑集团燕子山安置房项目施工围墙内。每天尤其上下班期间这条路上人流车流极多，安全隐患非常大。强烈请求文明城市 A 市，尽快整改这个极不文明的路段。	'A 市', '西湖', '建筑', '集团', '占', '道', '施工', '有', '安全隐患', 'A3 区', '大道', '西行', '便道', '未管', '所', '路口', '至', '加油站', '路段', '人行道', '包括', '路灯', '杆', '被', '圈', '西湖', '建筑', '集团', '燕子', '山', '安置', '房', '项目', '施工', '围墙', '内', '每天', '尤其', '上下班', '期间', '这', '条', '路上', '人流', '车流', '极', '多', '安全隐患', '非常', '大', '强烈', '请求', '文明城市', 'A 市', '尽快', '整改', '这个', '极', '不', '文明', '的', '路段'

（4）去除停用词

停用词是指对于模型的构建没有帮助，反而可能会产生不利的词语，因此需要去除。加载并扩充停用词表后，使用 apply()函数剔除表中所列出的全部停用词。

去除停用词后的部分结果如图所示：

表 2.2 去除停用词结果

Jieba 分词结果	去除停用词结果
'A 市', '西湖', '建筑', '集团', '占', '道', '施工', '有', '安全隐患', 'A3 区', '大道', '西行', '便道', '未管', '所', '路口', '至', '加油站', '路段', '人行道', '包括', '路灯', '杆', '被', '圈', '西湖', '建筑', '集团', '燕子', '山', '安置', '房', '项目', '施工', '围墙', '内', '每天', '尤其', '上下班', '期间', '这', '条', '路上', '人流', '车流', '极', '多', '安全隐患', '非常', '大', '强烈', '请求', '文明城市', 'A 市', '尽快', '整改', '这个', '极', '不', '文明', '的', '路段'	'A 市','西湖','建筑','集团','占','道', '施工', '安全隐患', 'A3 区', '大道', '西行', '便道', '未管', '路口', '加油站', '路段', '人行道', '包括', '路灯', '杆', '圈', '西湖', '建筑', '集团', '燕子', '山', '安置', '房', '项目', '施工', '围墙', '上下班', '期间', '条', '路上', '人流', '车流', '安全隐患', '文明城市', 'A 市', '整改', '文明', '路段'

（5）文本词汇向量化

为了对文本数据进行挖掘分析，可以将文本词汇以向量的形式表达出来。考虑到需要体现出生成词袋中的词频信息，本文采用 TF-IDF 算法，其基本原理为：

a.TF: Term frequency 即关键词词频，指一篇文档中关键词出现的频率。

$$TF = \frac{N}{M} \qquad \begin{matrix} N\text{：单词在某文档中的频次} \\ M\text{：该文档的单词数} \end{matrix} \qquad (1)$$

b.IDF: Inverse document frequency 指逆向文本频率，用于衡量关键词权重。

$$IDF = \log\left(\frac{D}{D_W}\right) \qquad \begin{matrix} D\text{:总文档数} \\ D_W\text{：出现了该单词的文档数} \end{matrix} \qquad （2）$$

c.TF-IDF:

$$TF\text{-}IDF = TF \times IDF \qquad （3）$$

Scikit-Learn 中 TF-IDF 权重计算方法主要用到两个类，使用 CountVectorizer 类将文本中的词语转换为词频矩阵，使用 TfidfTransformer 类统计 vectorizer 中每个词语的 TF-IDF 值。注意到测试集数据词语数量与训练集数据词语数量可能会相差很多，因此在实际分析时这里还需要进行一步维度共享操作。

词汇-文本词频矩阵的部分结果如图所示：

	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE
448	0	0	0	0	0	0	0	0	0	0	0
449	0	0	0	0	0	0	0	0	0	0	0
450	0	0	0	0	0	0	0	0	0	0	0
451	0	0	0	0	0	0	0	0	0	1	0
452	0	0	0	0	0	0	0	0	0	0	0
453	0	0	0	0	0	0	0	0	0	0	0
454	0	0	0	0	0	0	0	0	0	0	0
455	0	0	0	0	0	0	0	0	0	0	0
456	0	0	0	1	0	0	0	0	0	0	0
457	0	0	0	0	0	0	0	0	0	0	0
458	0	0	0	0	1	0	0	1	0	0	0
459	0	0	0	0	0	0	0	0	0	0	0
460	0	1	0	0	0	0	0	0	0	0	0
461	0	0	0	0	0	0	0	0	0	0	0
462	0	0	0	0	0	0	0	0	0	0	0
463	0	0	0	0	0	0	0	0	0	0	0
464	0	0	0	0	0	0	0	0	0	0	0
465	0	0	0	0	0	0	0	0	0	0	0

图 2.3 词汇-文本词频矩阵的部分结果

词汇-文本 TF-IDF 权重矩阵部分结果如图所示：

	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE
448	0	0	0	0	0	0	0	0	0	0	0
449	0	0	0	0	0	0	0	0	0	0	0
450	0	0	0	0	0	0	0	0	0	0	0
451	0	0	0	0	0	0	0	0	0	0.666384	0
452	0	0	0	0	0	0	0	0	0	0	0
453	0	0	0	0	0	0	0	0	0	0	0
454	0	0	0	0	0	0	0	0	0	0	0
455	0	0	0	0	0	0	0	0	0	0	0
456	0	0	0	0.643049	0	0	0	0	0	0	0
457	0	0	0	0	0	0	0	0	0	0	0
458	0	0	0	0	0.449166	0	0	0.658662	0	0	0
459	0	0	0	0	0	0	0	0	0	0	0
460	0	1	0	0	0	0	0	0	0	0	0
461	0	0	0	0	0	0	0	0	0	0	0
462	0	0	0	0	0	0	0	0	0	0	0
463	0	0	0	0	0	0	0	0	0	0	0
464	0	0	0	0	0	0	0	0	0	0	0
465	0	0	0	0	0	0	0	0	0	0	0

图 2.4 词汇-文本 TF-IDF 权重矩阵部分结果

2.1.3 留言内容分类

根据附件 1 提供的留言内容分类体系，使用支持向量机（support vector machines，SVM）找出与每一个一级分类标签相似的元素。SVM 是一个二分类的分类模型（或者叫做分类器）。它分类的思想是：给定给一个包含正例和反例的样本集合，寻找一个超平面来对样本根据正例和反例进行分割。其基本原理为：

（1）SVM 线性分类器

如果一个线性函数能够将样本完全正确的分开，就称这些数据是线性可分的，否则称为非线性可分的。在样本空间中，划分超平面可通过如下线性方程来描述：

$$g(x) = w^T x + b = 0 \tag{4}$$

SVM 的核心思想是尽最大努力使分开的两个类别有最大间隔，这样才使得分隔具有更高的可信度，而且对于未知的新样本才有很好的分类预测能力，SVM 的办法是：让离分隔面最近的数据点具有最大的距离。

假设完成了对样本的分隔，且两种样本的标签分别是{+1，-1}，那么对于一个分类器来说，$g(x) > 0$ 和 $g(x) < 0$ 就可以分别代表两个不同的类别，+1 和-1。为了描述离分隔超平面最近的数据点，需要找到两个和这个超平面平行和距离相等的超平面：

$$H_1: y = w^T x + b = +1 \text{ 和 } H_2: y = w^T x + b = -1 \tag{5}$$

在这两个超平面上的样本点也就是理论上离分隔超平面最近的点，决定了 H_1 和 H_2 的位置。这两个超平面定义了上面提到的间隔，二维情况下 $ax + by = c_1$ 和 $ax + by = c$ 两条

平行线的距离公式为：$\frac{|C_1-C_2|}{\sqrt{a^2+b^2}}$

由此可以推出 H1 和 H2 两个超平面的间隔为$\frac{2}{\|w\|}$，即要最大化这个间隔。这等价于最小化$\frac{1}{2}\|w\|^2$。假设超平面能将样本正确分类，则可令：

$$\begin{cases} w^T x_i + b \geq +1, y_i = +1 \\ w^T x_i + b \leq -1, y_i = -1 \end{cases} \quad (6)$$

两个式子综合有：$y_i(w^T x_i + b) \geq 1$

于是问题变为一个最优化问题：

$$\min \frac{1}{2}\|w\|^2 \quad (7)$$

$$suject\ to\ y_i(w^T x_i + b) - 1 \geq 0, (i = 1, 2, \cdots, j)(j\text{是样本数})$$

（2）核函数

上述方法无法完成对线性不可分样本的分隔，因此要把一个低维的样本集映射到高维，使它变成线性可分。

设映射函数为 Φ(•)，则映射后的空间分类函数变成：

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \langle \phi(x_i), \phi(x) \rangle + b \quad (8)$$

但是，如果拿到低维数据直接映射到高维的话，维度的数目会呈现爆炸性增长。所以这里需要引入核函数（kernal function）。

核函数的思想是寻找一个函数，使得在低维空间中进行计算的结果和映射到高维空间中计算内积$\langle \phi(x_1), \phi(x_2) \rangle$的结果相同。这样就避开直接在高维空间中进行计算，而最后的结果却是等价的。

分类函数：

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \, \mathrm{k}(x_1, x_2) + b \quad (9)$$

其中 k 是核函数。

（3）容错性： Outliers

由于噪音的存在，有可能有偏离正常位置很远的数据点存在，甚至类别 1 出现杂了

类别 2 的区域中这样的异常值叫 outliers。

为处理这种情况，SVM 允许数据点在一定程度上偏离超平面，约束就变成了:

$$y_i(w^T x_i + b) \geq 1 - \xi_i \tag{10}$$

其中 $\xi_i \geq 0$，称为松弛变量（slack variable），

引入容错性后，如果 ξ_i 任意大的话，那任意的超平面都是符合条件的，所以需要在原目标函数中加入损失函数：

$$\sum_{i=1}^{n} \xi_i \text{或} \sum_{i=1}^{n} \xi_i^2 \tag{11}$$

这时惩罚因子 C（cost）可帮助参数调优，它表示对离群点带来的损失的重视程度。惩罚因子的值越大，对目标函数的损失越大。

原来的优化问题就变成:

$$\min \ \frac{1}{2} \| w \|^2 + C \sum_{i=1}^{n} \xi_i \tag{12}$$

$$subject\ to\ y_i(w^T x_i + b) \geq 1 - \xi_i, (i = 1, 2, \cdots, j)(j\text{是样本数})$$

$$\xi_i \geq 0$$

（4）SVM 用于多类分类

SVM 算法最初是为二值分类问题设计的，当处理多类问题时，就需要构造合适的多类分类器。目前，构造 SVM 多类分类器的方法主要有两类 ：

a.直接法，直接在目标函数上进行修改，将多个分类面的参数求解合并到一个最优化问题中，通过求解该最优化问题“一次性”实现多类分类。这种方法看似简单，但其计算复杂度比较高，实现起来比较困难，只适合用于小型问题中；

b.间接法，主要是通过组合多个二分类器来实现多分类器的构造，常见的方法有 one-against-one 和 one-against-all 两种。

①一对多法（one-versus-rest,OVR SVMs）

训练时依次把某个类别的样本归为一类,其他剩余的样本归为另一类，这样 k 个类别的样本就构造出了 k 个 SVM。分类时将未知样本分类为具有最大分类函数值的那类。

②一对一法（one-versus-one,OVO SVMs 或 pairwise）

在任意两类样本之间设计一个 SVM，因此 k 个类别的样本就需要设计 k(k-1)/2 个

SVM。当对一个未知样本进行分类时，最后得票最多的类别即为该未知样本的类别。

本文所使用的方法为：一对多法。

2.2 问题二分析方法与过程 流程图

2.2.1 流程图

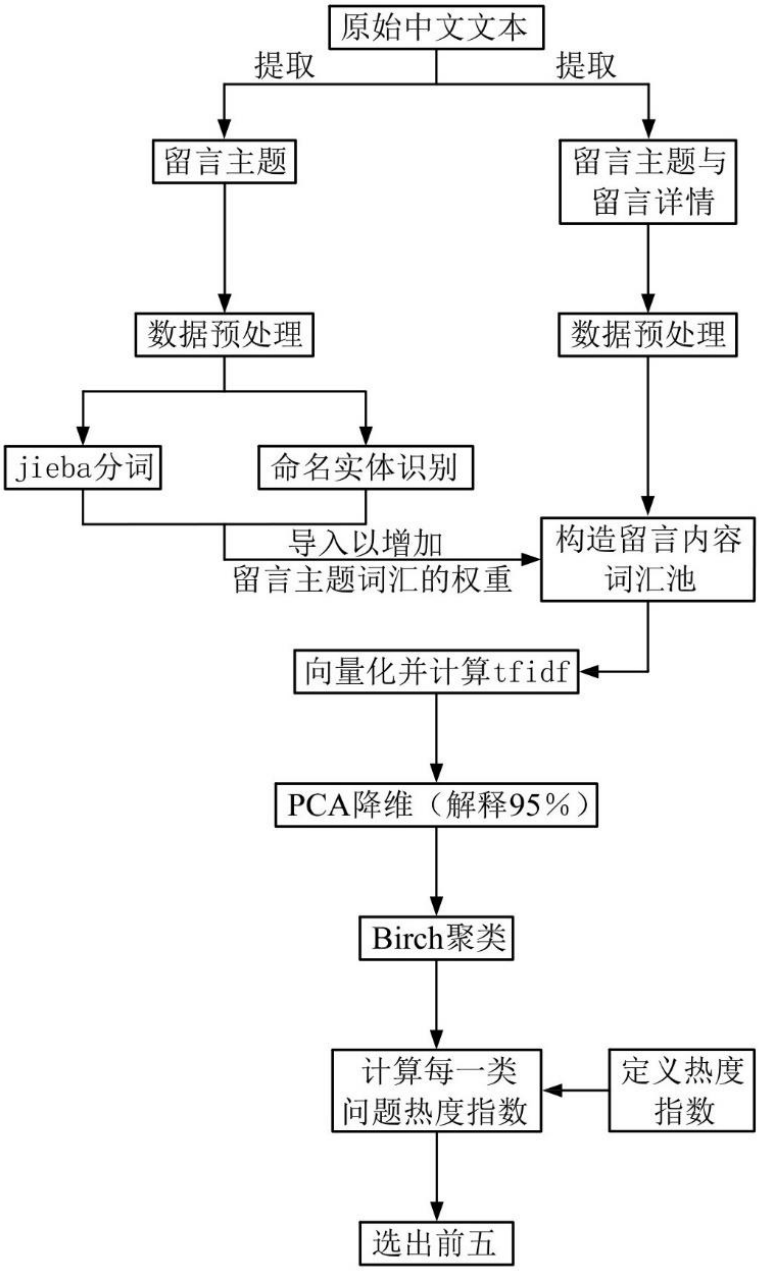

图 2.5 问题二流程图

2.2.2 数据预处理

（1）基本步骤

a.将合并留言主题和留言详情作为要分析的留言内容；

b.利用正则表达式选出与字母数字相关的关键词，增加个人词库；

c.去重、去敏、去除无意义符号；

d.中文分词并去除停用词；

e.利用 Hanlp 进行命名实体识别；

f.在留言内容中重复加入一遍留言主题分词以及命名实体识别的结果（此步骤目的为增加留言主题和命名题识别词的权重）。

（2）命名实体识别

命名实体识别（Named Entity Recognition，NER），又称作“专名识别”，是指识别文本中具有特定意义的实体，主要包括人名、地名、机构名、专有名词等。这里基于 HMM-Viterbi 算法，使用 hanlp 函数对分词后的留言内容进行命名实体识别。

隐式马尔可夫链模型（Hidden Markov mode，HMM）是一个五元组：

观测序列（observations）：实际观测到的现象序列

隐含状态（states）：所有的可能的隐含状态

初始概率（start_probability）：每个隐含状态的初始概率

转移概率（transition_probability）：从一个隐含状态转移到另一个隐含状态的概率

发射概率（emission_probability）：某种隐含状态产生某种观测现象的概率

一般地，HMM 模型可以用来解决三种问题：

a.参数(StatusSet, TransProbMatrix, EmitRobMatrix, InitStatus)已知的情况下，求解观察值序列。(Forward-backward 算法)

b.参数(ObservedSet, TransProbMatrix, EmitRobMatrix, InitStatus)已知的情况下，求解状态值序列。(viterbi 算法)

c.参数(ObservedSet)已知的情况下，求解(TransProbMatrix, EmitRobMatrix, InitStatus)。（Baum-Welch 算法)

维特比算法（Viterbi algorithm）是一种动态规划算法。它用于寻找最有可能产生观测事件序列的维特比路径——隐含状态序列，特别是在马尔可夫信息源上下文和隐马尔可夫模型中。其基本原理为：

假设给定隐式马尔可夫模型（HMM）状态空间 S，共有 k 个状态，初始状态 i 的概率为π_i，从状态 i 到状态 j 的转移概率为$a_{i,j}$。令观察到的输出为$y_1, y_2, ..., y_T$。产生观察结果的最有可能的状态序列$x_1, x_2, ..., x_T$由递推关系给出：

$$V_{1,k} = P\left(y_1 \mid k\right) \cdot \pi_k \tag{13}$$

$$V_{t,k}=P\left(y_t\mid k\right)\cdot\max_{x\in S}\left(a_{x,k}\cdot V_{t-1,x}\right) \tag{14}$$

此处$V_{t,k}$是前 t 个最终状态为 k 的观测结果最有可能对应的状态序列的概率。通过保存向后指针寄住在第二个等十种用到的状态 x 可以获得维特比路径。声明一个函数$Ptr\left(k,t\right)$，它返回 t>1 时计算$V_{t,k}$用到的 x 值或 t=1 时的 k。这样：

$$x_T=\arg\max_{x\in S}\left(V_{T,x}\right) \tag{15}$$

$$x_{t-1}=Ptr\left(x_t,t\right) \tag{16}$$

部分命名实体识别结果如图所示：

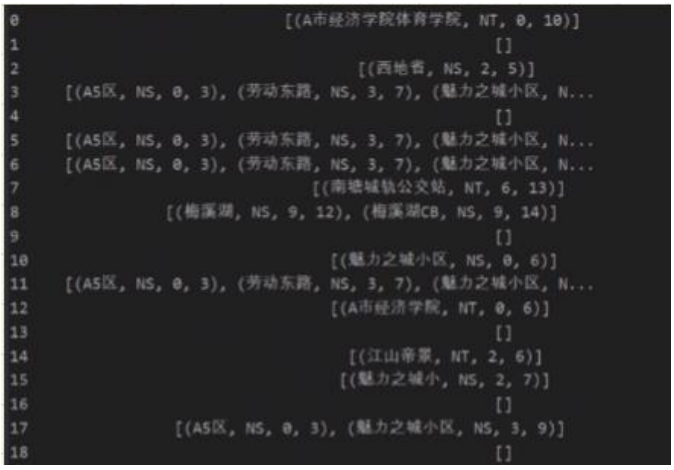

图 2.6 部分命名实体识别结果

2.2.3 PCA 降维

在理解特征提取与处理时，涉及高维特征向量的问题往往容易陷入维度灾难。随着数据集维度的增加，算法学习需要的样本数量呈指数级增加。另外，随着维度的增加，数据的稀疏性会越来越高。在高维向量空间中探索同样的数据集比在同样稀疏的数据集中探索更加困难。因此需要对预处理后的数据进行降维处理。本文使用 PCA（Principal Component Analysis）降维方法，又称为主成分分析。

PCA 可以把可能具有相关性的高维变量合成线性无关的低维变量，称为主成分，且新的低维数据集会尽可能的保留原始数据的变量。其基本原理为：

首先，对原始数据进行标准化处理：

$$x_{ik}^{'}=\frac{\left(x_{ik}-\overline{x_k}\right)}{S_k},i=1,2,...,n,k=1,2,...,p \tag{17}$$

式中，$\overline{x_k}=\sum_{i=1}^{n}x_{ik}/n,\mathrm{S}_k^{\,2}=\frac{1}{n-1}\sum_{i=1}^{n}\left(x_{ik}-\overline{x_k}\right)\left(x_{ik}-\overline{x_k}\right)^{'}$

标准化处理后，数据的方差为 1，均值为 0。

有 n 个样本，每个样本有 p 个变量，构成一个 $n\times p$ 的矩阵：

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} \tag{18}$$

对 X 进行线性变换，形成新的综合变量，用 Z 表示，记 $X_1,...,X_p$ 为原变量指标，$Z_1,...,Z_p$ 为新变量指标：

$$\begin{cases} \mathrm{Z_1=u_{11}X_1+u_{12}X_2+\cdots+u_{1p}X_p} \\ \mathrm{Z_2=u_{21}X_1+u_{22}X_2+\cdots+u_{2p}X_p} \\ \vdots \\ \mathrm{Z_p=u_{p1}X_1+u_{p2}X_2+\cdots+u_{pp}X_p} \end{cases} \tag{19}$$

上述线性变换的原则是：$u_{i1}{}^2+\cdots+u_{ip}{}^2=1$；$Z_i$ 与 Z_j 相互无关$(i\neq j;i,j=1,2,\cdots p)$；

Z_1 是一切满足 $X_1,...,X_p$ 所有线性组合中方差最大者，Z_2 是与 Z_1 不相关的 $X_1,...,X_p$ 所有线性组合中方差最大者，……，Z_p 是与 $Z_1,...,Z_{p-1}$ 都不相关的 $X_1,...,X_p$ 的所有线性组合中方差最大者，基于以上几条原则决定的综合变量 $Z_1,...,Z_p$ 称为原始变量的第一，第二，……，第 p 个主成分。其中，各综合变量在总方差中占的比重依次递减，通常情况下，选取的主成分个数要使得保持的信息总量的比重达到 85%以上即可。

因此，主成分分析的主要思想是对于原先提出的所有变量，将重复的变量（关系紧密的变量）删去多余，建立尽可能少的新变量，使得这些新变量是两两不相关的，而且这些新变量在反映课题的信息方面尽可能保持原有的信息。

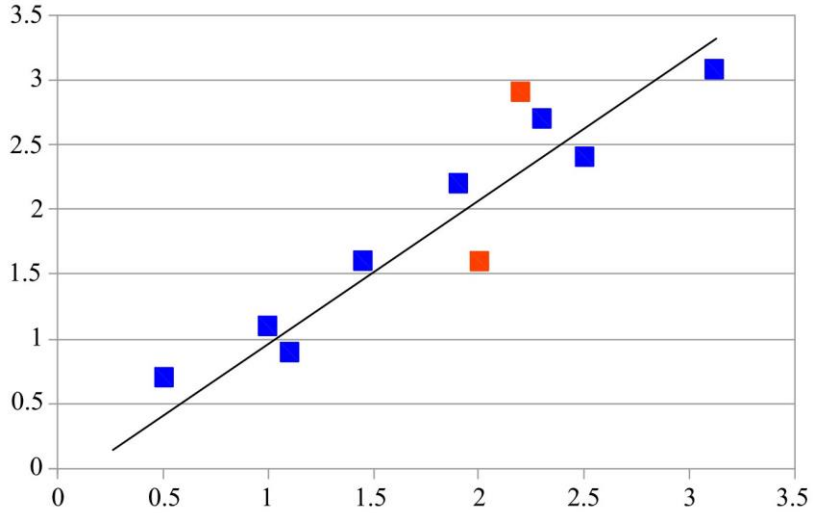

图 2.7 PCA 降维示意图

2.2.4 留言分类

为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,此处在生成职位描述的 TF-IDF 权重向量并进行降维后,根据每个职位的 TF-IDF 权重向量,对留言进行分类。这里采用的是 BIRCH 聚类方法。

BIRCH 算法比较适合于数据量大,类别数也比较多的情况,且运行速度很快,只需要单遍扫描数据集就能进行聚类。其基本原理为:

利用一个树结构来快速的聚类，一般称之为聚类特征树(CF Tree)。这棵树的每一个节点由若干个聚类特征(CF)组成。

一个 CF 是一个三元组，这个三元组就代表了簇的所有信息。给定 N 个 d 维的数据点$\{x_1, x_2, ..., x_n\}$，CF 定义如下：

$$CF = (\mathrm{N}, \mathrm{LS}, \mathrm{SS}) \tag{20}$$

其中，N 是子类中节点的数目，LS 是 N 个节点的线性和，SS 是 N 个节点的平方和。

CF 满足可加性，设$CF_1 = (N_1, LS_1, SS_1)$和$CF_2 = (N_2, LS_2, SS_2)$分别表示两个不相交的簇的聚类特征，如果将这两个簇合并成一个大簇，则大簇的聚类特征为

$$CF_1 + CF_2 = (N_1 + N_2, LS_1 + LS_2, SS_1 + SS_2) \tag{21}$$

此外，假如一个簇中包含 n 个数据点：$\{x_i\}, i = 1, 2, ..., n$，则可以计算簇的质心 C 和簇的半径 R：

$$C = (x_1 + x_2 + ... + x_n)/n \tag{22}$$

$$R=\left(\left|x_1-C\right|^2+\left|x_2-C\right|^2+...+\left|x_n-C\right|^2\right)/n \qquad (23)$$

其中，簇半径表示簇中所有点到簇质心的平均距离。CF 中存储的是簇中所有数据点的特性的统计和，所以当我们把一个数据点加入某个簇的时候，那么这个数据点的详细特征就丢失了。因此 BIRCH 聚类可以在很大程度上对数据集进行压缩。

CF 树的结构类似于一棵 B-树，它有三个参数：内部节点平衡因子 B，叶节点平衡因子 L，簇半径阈值 T。树中每个节点最多包含 B 个孩子节点，记为$\left(\mathrm{CF_i},\mathrm{CHILD_i}\right),1\le i\le B$，$CF_i$ 是这个节点中的第 i 个聚类特征，$\mathrm{CHILD_i}$ 指向节点的第 i 个孩子节点，对应于这个节点的第 i 个聚类特征。例如，一棵高度为 3，B 为 6，L 为 5 的一棵 CF 树的例子如图所示：

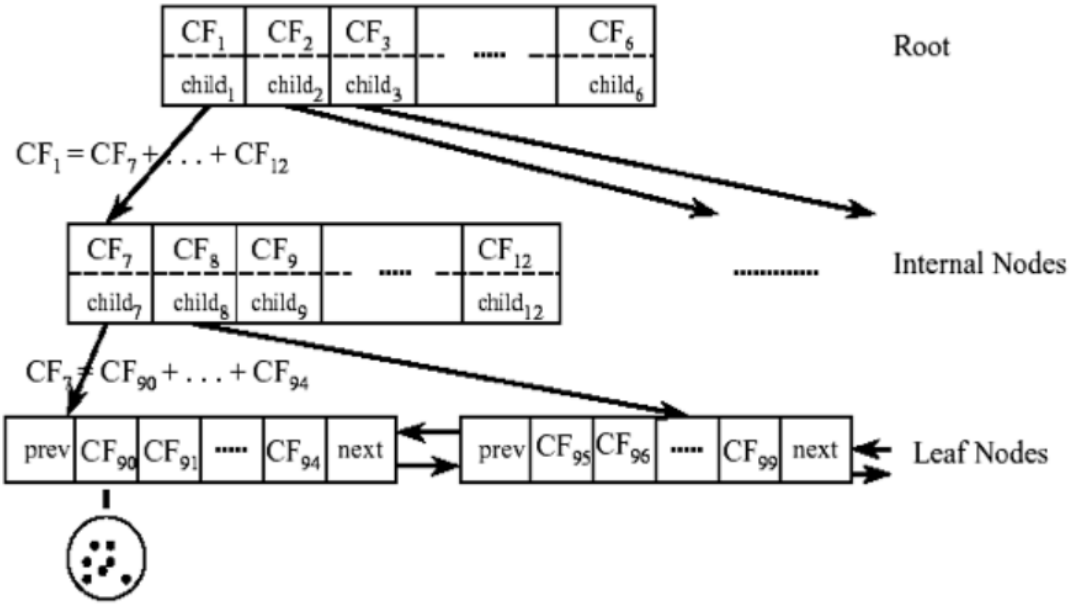

图 2.8 CF 树生成示意图

一棵 CF 树是一个数据集的压缩表示，叶子节点的每一个输入都代表一个簇 C，簇 C 中包含若干个数据点，并且原始数据集中越密集的区域，簇 C 中包含的数据点越多，越稀疏的区域，簇 C 中包含的数据点越少，簇 C 的半径小于等于 T。

CF 树的生长由一根空树开始，是逐个插入一系列簇的聚类特征到 CF 树的过程，基本步骤如下：

（1）从根节点向下寻找和新样本距离最近的叶子节点和叶子节点里最近的 CF 节点；

（2）如果新样本加入后，这个 CF 节点对应的超球体半径仍然满足小于阈值 T，则更新路径上所有的 CF 三元组，插入结束。否则转入 3；

（3）如果当前叶子节点的 CF 节点个数小于阈值 L，则创建一个新的 CF 节点，放入新样本，将新的 CF 节点放入这个叶子节点，更新路径上所有的 CF 三元组，插入结束。否则转入 4；

（4）将当前叶子节点划分为两个新叶子节点，选择旧叶子节点中所有 CF 元组里超球体距离最远的两个 CF 元组，分布作为两个新叶子节点的第一个 CF 节点。将其他元组和新样本元组按照距离远近原则放入对应的叶子节点。依次向上检查父节点是否也要分裂，如果需要按和叶子节点分裂方式相同。

BIRCH 聚类算法的基本步骤如下：

（1）将所有的样本依次读入，在内存中建立一颗 CF Tree, 建立的方法参考上一节。

（2）（可选）将第一步建立的 CF Tree 进行筛选，去除一些异常 CF 节点，这些节点一般里面的样本点很少。对于一些超球体距离非常近的元组进行合并。

（3）（可选）利用其它的一些聚类算法比如 K-Means 对所有的 CF 元组进行聚类，得到一颗比较好的 CF Tree.这一步的主要目的是消除由于样本读入顺序导致的不合理的树结构，以及一些由于节点 CF 个数限制导致的树结构分裂。

（4）（可选）利用第三步生成的 CF Tree 的所有 CF 节点的质心，作为初始质心点，对所有的样本点按距离远近进行聚类。这样进一步减少了由于 CF Tree 的一些限制导致的聚类不合理的情况。

2.2.5 热度排名

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。这里对每一类问题的热度进行排名。方法如下：

（1）统计每一类问题的留言次数。考虑到同一个问题可能会有同一个用户多次反映，这里将其视为 1 次有效反映。

（2）计算每一问题的热度指数：

$$\text{热度指数}=\frac{\text{问题的留言次数}}{\text{留言最多问题的留言次数}}\times 10 \qquad (24)$$

（3）选出热度指数最高的 5 个问题。

2.3 问题三分析方法与过程

答复意见的质量评价包括：答复内容本身的质量评价（包括答复的完整性）、问题和答复的匹配度（包括答复的相关性、可解释性）计算。

其中，答复内容本身的质量评价可以包括以下内容中的至少一种：1、答复的长度，根据统计长度适中的答复通常具有较高的质量；2、答复中视觉特征信息，包括：对于每个段落词个数，段落词首是否具有黑体加重符号等等，通常质量较高的答复除了长度适中以外，答复也有很好的视觉特征信息。3、答复正反例词典特征，即答复中的词在正例词典和反例词典中的比例。

由此，本文提出了一套完整的答复内容质量评价方案。

3.结果分析

3.1 问题一结果分析

3.1.1 留言分类模型

本文基于数据清洗技术、TF-IDF 权重策略及支持向量机（Support Vector Machine, SVM）多分类算法等，建立了关于留言内容的一级标签分类模型。基本步骤如下：

（1）合并留言主题和留言详情，作为要分析的留言内容；

（2）对留言内容去重、去敏、去除无意义符号；

（3）利用正则表达式增加个人词库；

（4）使用 python 中的 jieba 分词工具对留言内容进行中文分词；

（5）基于 TF-IDF 权重策略构建词汇矩阵；

（6）使用支持向量机（Support Vector Machine, SVM）多分类算法对留言内容进行一级标签分类。

3.1.2 分类方法评价

为确定最优分类方法，本文先是分别使用了多个分类方法对留言进行分类，然后使用 F-Score 对每种分类方法进行评价，所有评价结果见下表：

表 3.1 分类方法评价表

分类方法	F1-score
GaussianNB	0.658759
MultimomialNB	0.680917
DecisionTree	0.766450
RandomForest	0.850742
ExtraTrees	0.857799
KNN	0.853564
LogisticRegression	0.888566
XGBoost	0.869819
AdaBoost	0.727424

GradientBoosting	0.853869
SVM	**0.925953**

可以发现，SVM 算法的 F-Score 得分最高。这得益于 SVM 算法具有以下特点：

(1)非线性映射是 SVM 方法的理论基础,SVM 利用内积核函数代替向高维空间的非线性映射。

(2)对特征空间划分的最优超平面是 SVM 的目标,最大化分类边际的思想是 SVM 方法的核心。

(3)支持向量是 SVM 的训练结果,在 SVM 分类决策中起决定作用的是支持向量。

(4)SVM 是一种有坚实理论基础的新颖的小样本学习方法。它基本上不涉及概率测度及大数定律等,因此不同于现有的统计方法。从本质上看,它避开了从归纳到演绎的传统过程,实现了高效的从训练样本到预报样本的“转导推理”,大大简化了通常的分类和回归等问题。

(5)SVM 的最终决策函数只由少数的支持向量所确定,计算的复杂性取决于支持向量的数目,而不是样本空间的维数,这在某种意义上避免了“维数灾难”。

(6)少数支持向量决定了最终结果,这不但可以帮助我们抓住关键样本、“剔除”大量冗余样本,而且注定了该方法不但算法简单,而且具有较好的“鲁棒”性。这种“鲁棒”性主要体现在:

①增、删非支持向量样本对模型没有影响

②支持向量样本集具有一定的鲁棒性

③有些成功的应用中,SVM 方法对核的选取不敏感

(7)SVM 学习问题可以表示为凸优化问题，因此可以利用已知的有效算法发现目标函数的全局最小值。而其他分类方法（如基于规则的分类器和人工神经网络）都采用一种基于贪心学习的策略来搜索假设空间，这种方法一般只能获得局部最优解。

(8)SVM 通过最大化决策边界的边缘来控制模型的能力。尽管如此，用户必须提供其他参数，如使用核函数类型和引入松弛变量等。

(9)SVM 在小样本训练集上能够得到比其它算法好很多的结果。支持向量机之所以成为目前最常用，效果最好的分类器之一，在于其优秀的泛化能力，这是是因为其本身的优化目标是结构化风险最小，而不是经验风险最小，因此，通过 margin 的概念，得到对数据分布的结构化描述，因此减低了对数据规模和数据分布的要求。SVM 也并不是在任何场景都比其他算法好，对于每种应用，最好尝试多种算法，然后评估结果。如 SVM

在邮件分类上，还不如逻辑回归、KNN、bayes 的效果好。

(10)它基于结构风险最小化原则，这样就避免了过学习问题，泛化能力强。

(11)它是一个凸优化问题，因此局部最优解一定是全局最优解的优点。

(12)泛华错误率低，分类速度快，结果易解释。

3.2 问题二结果分析

本文使用数据清洗、命名实体识别、PCA 降维和 BIRCH 聚类等算法将附件 3 中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，并定义了合理的热度评价指标：

$$\text{热度指数}=\frac{\text{问题的留言次数}}{\text{留言最多问题的留言次数}}\times 10$$

最终给出排名前 5 的热点问题。

表 3.2 热点问题表

问题描述	热度指数	排名
A 市 A2 区万家丽南路丽发新城居民区附近搅拌站扰民	10.0	1
A 市武广新城伊景园滨河苑捆绑销售车位	9.730	2
A 市人才新政咨询	5.856	3
A 市 A4 区 58 车贷案件进展	4.324	4
A7 县星沙四区凉塘路的旧城改造	4.144	5

详细结果见附件“热点问题表.xlsx” 和“热点问题留言明细表.csv”。

3.3 问题三结果分析

答复意见的质量评价包括：答复内容本身的质量评价（包括答复的完整性）、问题和答复的匹配度（包括答复的相关性、可解释性）计算。

其中，答复内容本身的质量评价可以包括以下内容中的至少一种：1、答复的长度，根据统计长度适中的答复通常具有较高的质量；2、答复中视觉特征信息，包括：对于每个段落词个数，段落词首是否具有黑体加重符号等等，通常质量较高的答复除了长度适中以外，答复也有很好的视觉特征信息。3、答复正反例词典特征，即答复中的词在正例词典和反例词典中的比例。

针对问题三的要求，本文提出了一个完整的解决方案，如下：

为反映答复内容本身的质量，需要定义正例词典和反例词典。如果答复中的词在正

例词典中的比例较大，则该答复作为高质量的可能性较高；反之，如果答复中的词在正例词典中的比例较大，则该答复作为高质量的可能性较低。

正例词典和反例词典的构建过程如下：首先，提取大量问答对（如 5000 个）的语料，并将其标注两类，一类为高质量数据集 D1，另一类为中低质量数据集 D2；对提取的问题和答复中出现的所有词汇进行统计，如果某个词汇在高质量数据集 D1 中的频率除以在整个数据集（包括 D1 和 D2) 中的频率大于预定的阈值 a1, 则该词汇进入正例词典；如果某个词汇在高质量数据集 D1 中的频率除以在整个数据集（包括 D1 和 D2) 中的频率小于预定的阈值 a2, 则该词汇进入反例词典。问题中出现的词汇进入问题的正反例词典，答复中出现的词汇进入答复的正反例词典。

本小组提出一种基于主题聚类的方法进行问题和答复的匹配度计算，具体为：

步骤 1，收集一定量的全局语料库作为点互信息的统计语料，对该统计语料进行分词处理，并根据以下公式计算词与词之间的点互信息量。

$$\mathrm{PMI}(w_1,w_2)=\log_2\frac{P(w_1,w_2)}{P(w_1)P(w_2)} \tag{25}$$

其中，$\mathrm{PMI}(w_1,w_2)$表示词w_1与词w_2之间的点互信息量，$P(w_1)$表示词w_1在统计数据中的出现频率，$P(w_2)$表示词w_2在统计数据中的出现频率，$P(w_1,w_2)$表示词w_1和w_2的共现频率，即如果词w_1和w_2出现在连续几个句子中，且这连续的几个句子的字数小于长度阈值（如 150 个汉字），则认为词w_1和w_2共现。另外，在一个文档中w_1和w_2出现多次的，均只计算一次。

步骤 2，对簇内的问题进行分词和词性标注等处理，保留具有名词词性的词汇$q_1q_2,\dots,q_m$名词的个数记为 m。

步骤 3，对答复进行处理，判断答复的长度，如果大于长度阈值（如 150 个汉字），则对其进行主题词提取处理，主题词提取的主要操作为：从全局统计语料中查找答复的每个词所出现的词频 tf 和文档频率 df，并采用公式为每个词赋上权值：

$$\mathrm{TF}\log(\mathrm{tf})\log\left(\frac{1}{\mathrm{df}}\right) \tag{26}$$

按照权值从大到小的顺序对答复中的所有词进行排序，并提取靠前的若干个（例如 n=50) 名词作为主题词。其中 TF 表示对应的词在其所在的答复中所统计的局部频率。如果答

复的长度小于长度阈值，则直接对其进行分词、词性标注等处理，并提取具有名词词性的词汇$a_1 a_2, \ldots, a_n$个数记为 n

步骤 4，以 qi 为主题初始点，判断 aj 与 qi 的点互信息 是否大于点互信息阈值，如果大于，则将 aj 加入中心链；如果均小于点互信息阈值，则将 aj 删除。最终得到中心链中包含的词汇个数记为 k, 定义问题与答复之间的匹配度为：k/(m+n)。该定义表示，如果答复中的关键词和问题中关键词相关的越多，该概率就越大，表示提问和答复的相关度越高。

此外，为了融合上述多种特征，本小组采用最大熵统计模型作为各个特征的融合框架，以实现对单个问答对质量的评价。当然，本小组中的融合框架也可以采用其他类型的分类器来实现，例如：支持向量机、贝叶斯等，且本小组的融合框架并非仅限于上述所举。

下面以最大熵评价分类器为例对各个特征的融合过程进行详细阐述，如图所示，最大熵评价分类器采用的输入特征包括：答复的长度、答复中视觉特征信息、答复正反例词典特征、问题和答复的匹配度。

其中，答复正反例词典特征的产生过程为：分别统计答复中的每个词在正反例词典中属于高质量数据和属于低质量数据的概率；然后利用贝叶斯公式计算得到 P(good|A) 的概率，该概率作为答复正反例词典特征的输入。

答复的长度定义为该长度 L，下属于高质量数据的概率$p(good|L)$ ，且

$$\mathrm{P}(\mathrm{good}\,|\,\mathrm{L})=\frac{\mathrm{P}(\mathrm{good})\mathrm{p}(\mathrm{good}\,|\,\mathrm{L})}{\mathrm{P}(\mathrm{good})\mathrm{p}(\mathrm{good}\,|\,\mathrm{L})+\mathrm{P}(\mathrm{bad})\mathrm{p}(\mathrm{bad}\,|\,\mathrm{L})} \qquad (27)$$

概率$p(good|L)$，$p(bad|L)$是在训练过程中进行统计得到的。

答复中视觉特征信息是根据判断最终形成是否满足格式化信息所得到的结果，如果满足，则该特征信息为 1，否则为 0。

上述的训练过程为，首先在 5000 个训练样本中训练出最大熵的模型参数，然后利用最大熵的模型参数进行识别，最终给每个问答对有一个正确的评价分值，以此作为问答对内的质量评价结果。对于分值低于一定阈值的问答对则认为是中低质量的问答对，直接删除。

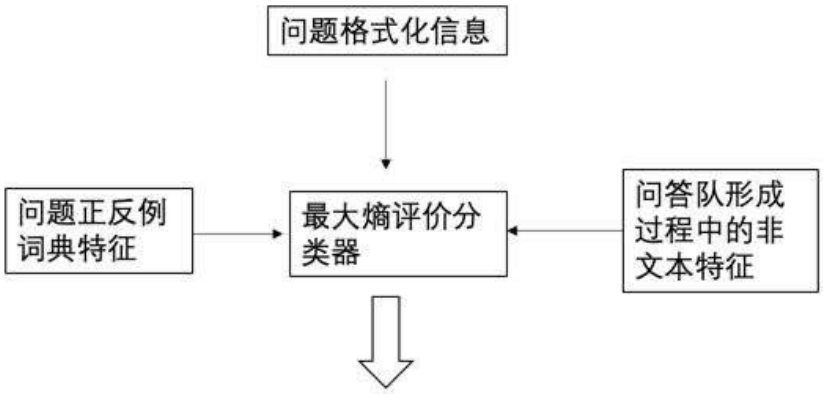

图 3.1 答复内容评价方案流程图