“智慧政务”中的文本挖掘应用

摘要： 近年来，随着网络问政平台的普及，各类民意调查的文本数据量不断攀升，给过去依靠人工来进行留言划分和热点整理的相关部门带来了极大挑战。因此，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，对提升政府的工作效率具有极大的推动作用。

本文将基于收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，利用自然语言处理和文本挖掘的方法解决群众留言分类、热点问题挖掘和答复意见评价三个问题。

针对问题一：将数据进行预处理后，然后基于 TF-IDF 算法构建词向量。以及利用奇异值分解算法(SVD)对矩阵进行降维。最后通过支持向量机模型(SVM)进行分类。

针对问题二：与问题一类似，先进行预处理并降维，用 BIRCH 聚类算法对留言问题进行聚类，然后改进 Reddit 的排列推送算法作为热度算法计算出热点问题，再基于 TextRank 算法对热点问题进行摘要，并用 CRF 条件随机场模型对热点问题进行命名实体识别并进行优化。

针对问题三：要求是设计一套对留言答复意见的评价方案。本题从相关性、可解释性做出解决方法。

相关性，即答复意见内容与本题相关，本题采用了相似度去求取相关性。使用了欧式距离和余弦相似度公式计算。

可解释性，即答复意见中内容的相关解释。答非所问的回复被认为相关性低。使用了朴素贝叶斯分类器。

关键词：中文分词；TF-IDF；SVD；支持向量机；BIRCH聚类；文本摘要；命名实体识别；文本相关性

Abstract

Abstract: In recent years, with the popularity of online political platform, the amount of text data of all kinds of public opinion surveys keeps increasing, which brings great challenges to relevant departments that used to rely on people to divide comments and sort out hot topics. Therefore, the establishment of intelligent government system based on natural language processing technology has been a new trend of social governance innovation and development, which plays a great role in promoting the work efficiency of the government.

This paper will be based on the records of political comments collected from relevant departments to the comments of some people, the author uses the methods of natural language processing and text mining to solve three problems, namely, the classification of comments, hot issues mining and the evaluation of replies. According to the first problems: after preprocessing the data, construct the word vector based on TF-IDF algorithm. And the singular value decomposition algorithm (SVD) is used to reduce the dimension of the matrix. Finally, support vector machine model (SVM) was used for classification.

To solve the second problem: with a similar problem, first preprocessing and dimension reduction, then, to leave a message with BIRCH clustering algorithm for clustering, then TextRank algorithm based on hot issues in this paper, and the CRF conditional random field algorithm to named entity recognition of hotspot issues, finally improve the arrangement of Reddit push algorithm, formation heat calculation algorithms.

For the third problem: the requirement is to design a set of comments on the comments of the evaluation program. This topic from relevance, interpretability to make a solution.

Relevance, that is, the content of the answer is related to the topic, the topic uses The Euclidean distance and cosine similarity formulas to find relevance.

Interpretability refers to the relevant interpretation of the content of the replies. Irrelevant answer responses are considered to be of low relevance.

Key words: Chinese word segmentation; TF - IDF; SVD; Support vector machine; BIRCH clustering; text summarization; Named Entity Recognition; Textual relevance

目录

1 群众留言分类

1.1 总体流程

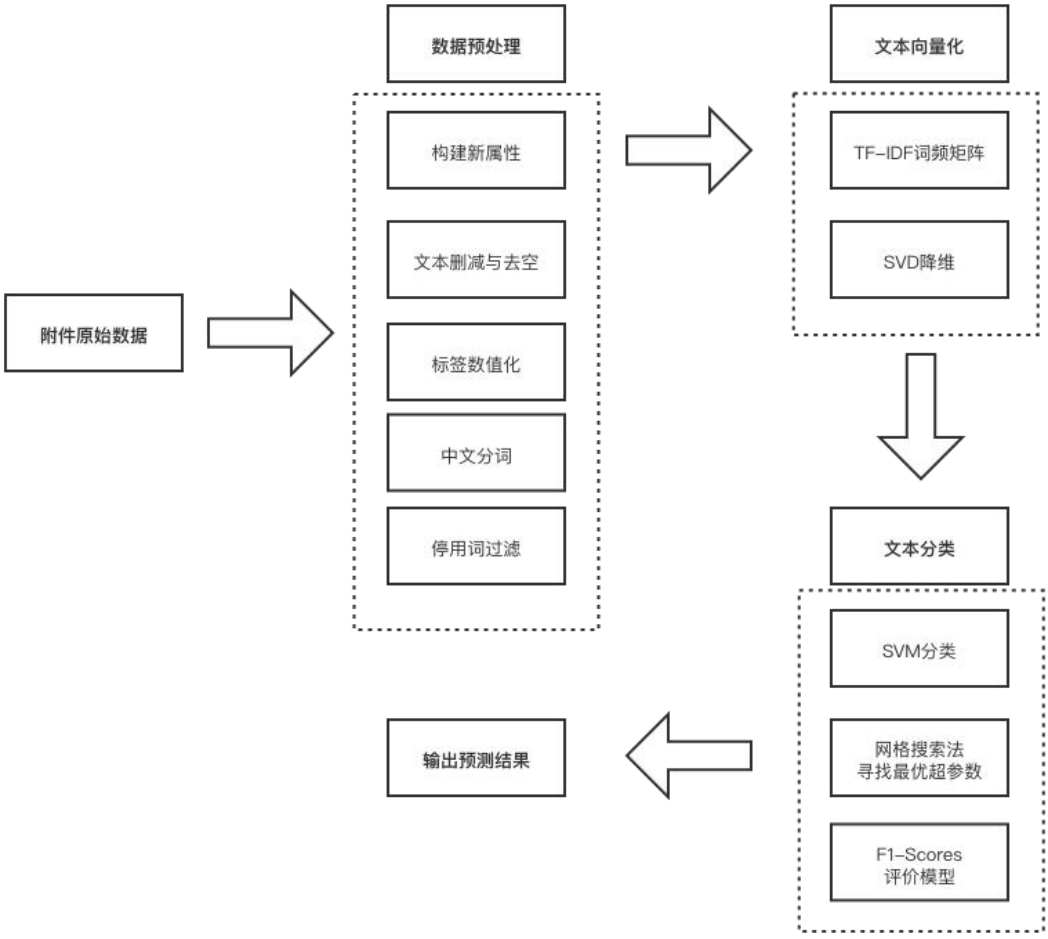


图 1.1 总体流程图

1.2 数据预处理

1.2.1 数据描述

通过分析题目，得出所需数据文件为附件 2.xlsx。观察所给数据，发现共有 9210 个样例,每个样例由 5 个属性描述(留言编号，留言用户，留言主题，留

言时间，留言详情)和 1 个标签(一级标签)组成。而属性(留言详情)所对应的属性值中有大量的换行符和制表符以及没有意义的语句，如果不做处理会对后续分析造成影响。此外标签的值(城乡建设，环境保护，交通运输，教育文体，劳动和社会保障，商贸旅游，卫生计生)为文本格式，需要将其量化成数值形式才能对其进行预测。于是首先要对数据进行预处理。

1.2.2 文本预处理

(一)构建新的属性

由于留言主题的概括性强但留言详情又包含很多细节，于是将留言主题复制成两条-表示权重为 2，与留言详情拼合成新的属性(留言)。

(二)文本删减与去空

对于留言详情所对应的属性值，例如：

’\n\t\t\t\t\t\n\t\t\t\t\tA3 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。\n\t\t\t\t\t\n\t\t\t\t\t\n\t\t\t\t\t’

经过测试得出本题数字和英文字母对分类结果并没有积极意义，于是将数字和字母与’\t’、’’\n’、’\u3000’以及空格、标点符号去除

(三)标签数值化

使用 sklearn. preprocessing 的 LabelEncoder()函数将文本标签转化为数字

1. 关于 sklearn 库

Scikit-learn(sklearn)是机器学习项目开发中常用的第三方库，对大量机器学习工具和方法进行了封装，具有简单高效的特点。

2. 关于 LabelEncoder()方法

LabelEncoder()是 sklearn 的预处理包 preprocessing 中的方法，它将标签的值映射为在 0～(标签种类数-1)中的连续自然数

(四)中文分词

由于中文与英文在词与词之间的构成有很大的区别。英文句子中每个词之间都由明显的分割符如空格和标点符号，而中文则是词与词之间没有明显的分割符。这种结构上的不同导致的结果就是文本处理时要使用两种截然不同的方式，英文按照分隔符就可以很轻松的分词，而中文则需要以特定的方式将字符串进行切割，形成多个子串也就是词，此外很多时候还需要对词性做标记。

本题所用 python 的第三方库-jieba 分词，对属性(留言)的每一个值进行分词。jieba 库涉及的算法如下：

1. 基于前缀词典实现词图扫描,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)，采用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；

2. 对于未登录词，采用了基于汉字成词能力的 HMM 模型，采用 Viterbi 算法进行计算；

3. 基于 Viterbi 算法的词性标注；

4. 分别基于 TF-IDF 和 TextRank 模型抽取关键词；

jieba 分词系统主要包括三个模块：分词、词性标注、关键词抽取。

以下是本题分词结果示例：

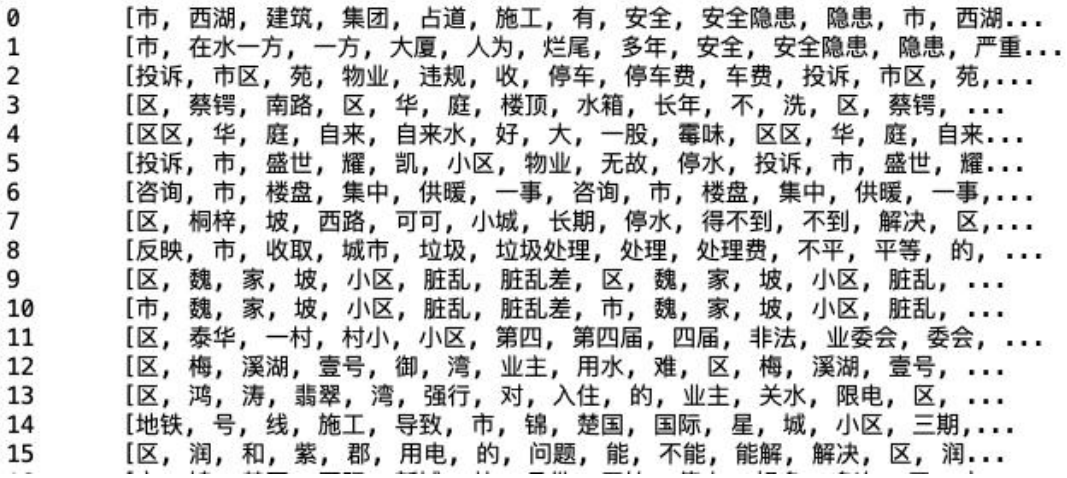


图 1.2.1 分词结果示例

(五)停用词过滤

在文本中有很多代词、助词、语气词等没有意义的词(我、你、他、虽然、但是、了 …)还包括标点符号和其他非汉字符号，这些字符统称为停用词，保留会增加模型计算时间甚至导致过拟合影响到模型效果。所以需要使用停用词典进行过滤。停用词典部分如下：

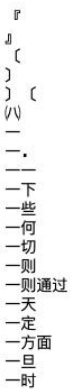


图 1.2.2 停用词典(部分)

1.3 文本空间向量模型

1.3.1 词向量

即便是在分词后的文本，仍然是又词构成，要想要计算机能够识别并计算，需要将文本转换为可识别的数据(词向量)

1.3.2 词频矩阵

将表内每一行的字符串作为一个文档，将所有文档中的所有词存入一个字典中，叫词典索引，其形式为：{word1:index, word2:index2}

```
Out[26]: {'南华': 22586,
          '培训': 30054,
          '学校': 33362,
          '利用': 18763,
          '周末': 26892,
          '时间': 53905,
          '违规': 83918,
          '公立': 15800,
          '公立学校': 15811,
          '学生': 33690,
          '文化': 52229,
          '文化课': 52342,
          '领导': 88641,
          '教育': 51354,
          '教育部': 51685,
          '明文': 54155,
          '明文规定': 54163,
          '校外': 57585,
          '机构': 56227,
```

图 1.3.2 词典索引

词频矩阵可以将文本数字化并加上词频表示成为一个矩阵：

$$\mathrm{V}(d) = [[(D_1,I_1), C_1],[(D_2,I_2), C_2],[(D_3,I_3), C_3],\ldots]$$

其中，$D_i(i = 1,2,3,\ldots,n)$为一个词所出现的文档号，$I_i(i = 1,2,3,\ldots,n)$为词在词典索引中所对应的索引，$C_i(i = 1,2,3,\ldots,n)$为词在该文档中出现的词频。 $C_i(i = 1,2,3,\ldots,n)$也可以为该特征值(词)的权值，可以使用 TF-IDF 算法计算得出。

```
print(vectorizer.fit_transform(docs))
  (0, 3)        1
  (0, 14)       1
  (0, 0)        1
  (0, 12)       1
  (1, 2)        1
  (1, 10)       1
  (1, 8)        1
  (2, 4)        1
  (2, 11)       1
  (2, 5)        1
  (2, 9)        1
```

图 1.3.3 词频矩阵

1.3.3 TF-IDF

TF-IDF(Term Frequency-Inverse Document Frequency/词频-逆文本频率)是中文文本处理中常用的计算特征权重的方法，可以分成两个部分 TF 和 IDF。TF 表示词频，统计文本中每个词出现的概率。IDF 表示逆文本概率，反映词语在文本中的重要性[1]。一般来说一个词的 IDF 会随着他出现的文档次数越多而下降。相反，一个词在较少的文档中出现，则 IDF 会增高。举个特例“电脑”一次在所有文档中都各出现一次则 IDF 值为 0，表明这个词在文本中很不重要。IDF 计算公式如下：

$$\mathrm{IDF}(x) = \log\frac{N}{N(x)} \#(1)$$

其中 N 为所有文档个数，N(x)表示含有 x 的文档个数。

根据上述公式(1)，就可以得到 TF-IDF 的计算公式：

$$\mathrm{TF} - IDF = TF(x) \times IDF(x) \#(2)$$

其中 TF(x)表示 x 在当前文档中的词频。

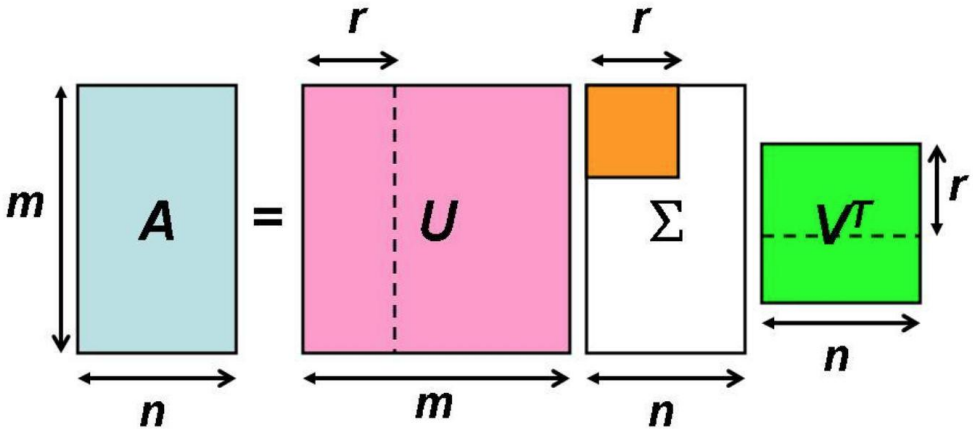
1.3.4 空间降维

1.3.4.1 SVD 奇异值分解理论

对于方阵可以使用特征值分解提取矩阵特征。但在现实和中文文本处理中大多数矩阵都不是方阵。例如通过提取文本 TF-IDF 特征得到的词频矩阵行数 M 表示有 M 个文档以及列数 N 表示语料库中有多少种词，这样形成的 M*N 的矩阵很可能不是方阵，于是可以使用 SVD 对矩阵进行分解但是和特征分解不同，SVD 并不要求要分解的矩阵为方阵。假设矩阵 A 是一个 $m \times n$ 的矩阵，那么定义矩阵 A 的 SVD 为：

$$A = U\sum V^T \#(3)$$

其中 U 是一个 $m \times m$ 的矩阵，Σ 是一个 $m \times n$ 的矩阵，其非主对角线的元素都是 0，主对角线上的每个元素称为奇异值，V 是一个 $n \times n$ 的矩阵。U 和 V 都是满足满足$U^TU = I, V^TV = I$ 的酉矩阵，而公式(3)就是 A 的奇异值分解[2]。如下图所示，r 为矩阵 A 的秩：



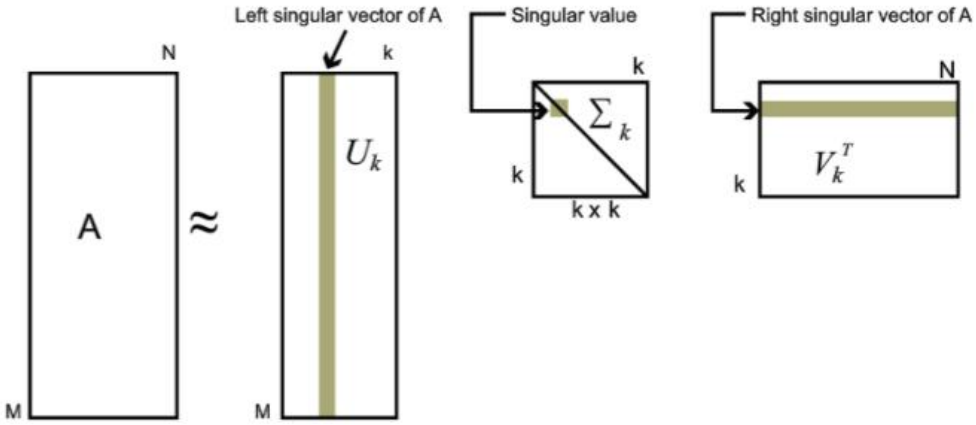
图片来源：慕课网 http://www.imooc.com/article/267351?block_id=tuijian_wz

1.3.4.2 SVD 的性质

上文所提到的奇异值,与特征分解中的特征值类似，在奇异值矩阵中奇异值的排序是从大到小的顺序，此外奇异值数值的下降特别的快。在很多时候，前十分之一的奇异值之和就约等于全部奇异值之和。因此使奇异值具有可压缩降噪的性质，可以用最大的 k 个的奇异值和对应的左右奇异向量来近似描述矩阵。表示如下：

$$A_{m\times n} = U_{m\times m}\sum_{m\times n}V^T_{n\times n} \approx U_{m\times k}\sum_{k\times k}V^T_{k\times n}$$

其中 k 要比 n 小很多，也就是一个大的矩阵 A 可以用三个小的矩阵 $U_{m\times k},\sum_{k\times k},V^T_{k\times n}$来表示。如下图所示，大矩阵 A 只需要灰色的部分的三个小矩阵就可以近似描述:



图片来源: https://www.cnblogs.com/pinard/p/6251584.html

由于这个重要的性质，SVD 可以用于 PCA 降维，来做数据压缩和去噪。也可以用于推荐算法，得到隐含的用户需求来做推荐。同时也可以用于 NLP 中的算法对词频矩阵进行压缩和降噪。

1.3.4.3 SVD 降维实现

本项目使用了 sklearn 库中的 TruncatedSVD 方法实现 SVD 算法来进行数据的降维，由于词频矩阵是一个非常稀疏的矩阵，每一行都有大量的 0 所有在降维前数据训练集的维度为：

```
xtrain_tfv.shape #降维前数据的维度
```

```
(8289, 90364)
```

图 1.3.4.1 训练集数据降维前的维度

其中 8289 表示训练集有 8289 个文档也就是行数；90364 表示维度。

调用函数将目标维度参数 n_components 设为 150，

```
#使用SVD进行降维，components设为150，对于SVM来说，SVD的components的合适调整区间一般为120~200
svd = decomposition.TruncatedSVD(n_components=150, random_state=42)
svd.fit(xtrain_tfv)
xtrain_svd = svd.transform(xtrain_tfv)
xvalid_svd = svd.transform(xvalid_tfv)
```

图 1.3.4.2 SVD 降维实现

降维后的矩阵大小为：

```
xtrain_svd.shape
```

```
(8289, 150)
```

可以看到维度有了很大程度的减少从原来的 90364 降为 150，在后续测试中可以发现这种降维与未降维相比对最终模型效果没有太多影响甚至提高了模型的效果，但大大减少了模型的计算时间。下面使用少量数据来证实这个结果：

```
未降维运行时间: 1.4753240000000005 Seconds
```

图 1.3.4.3 未使用 SVD 降维模型运行时间

```
print('降维后运行时间: %s Seconds'%(end-start))
```

```
降维后运行时间: 0.7616620000000012 Seconds
```

图 1.3.4.4 使用 SVD 降维后模型运行时间

其中考虑到使用全部数据运行时间过长，选择使用了 400 条文档进行测试。结果表明在少量数据的情况下，未降维是降维后运行时间的两倍。这一现象会随着数据的增加越发明显。

1.4 文本分类

1.4.1 基于词向量的支持向量机的文本分类

1.4.1.1 支持向量机(SVM)原理

支持向量机(Support Vector Machine，简称 SVM)于 1995 年正式发表，由于其在文本分类任务中具有很少过度拟合，适合解决训练样本少、特征维数高的的特点，很快成为机器学习的主流技术[3]。

SVD 的思想是基于训练集 D 在样本中找到具有“最大间隔”的划分超平面，将不同类别的文本数据分开。[4]

对于线性可分问题优化函数方程为：

$$\min_{w,b}\frac{1}{2}\|w\|^2$$

$$s.t.\ y_i(w^T x_i + b) \geqslant 1,\ i = 1,2,\ldots,m.\#(4.1)$$

其中 w,b 为超平面(w, b)的参数，可以看出间隔似乎仅与 w 有关，但实际上 b 通过约束隐式地影响着 w 的取值，从而对间隔产生影响。

对与非线性可分问题，通过核函数 $K(x_i, x_j)$将训练样本映射到一个更高维的特征空间。从而可在高维特征空间中构造一个分类间隔最大的超平面[5]

常见的核函数如下：

名称	表达式	参数
线性核	$k(x_i,x_j)=x_i^T x_j$	
多项式核	$k(x_i,x_j)=(x_i^T x_j)^d$	$d \geqslant 1$ 为多项式的次数
高斯核(RBF)	$k(x_i,x_j)=\exp\left(-\frac{\|x_i-x_j\|^2}{2\sigma^2}\right)$	$\sigma>0$ 为高斯核的带宽
拉普拉斯核	$k(x_i,x_j)=\exp\left(-\frac{\|x_i-x_j\|}{\sigma}\right)$	$\sigma>0$
Sigmoid 核	$k(x_i,x_j)=\tanh(\beta x_i^T x_j+\theta)$	tanh 为双曲正切函数，$\beta>0,\theta<0$

1.4.1.2 SVM 参数选择

本题使用 sklearn 的 SVC()方法实现支持向量机分类，其中对模型影响最大的参数是：惩罚因子 C，和 gamma 参数。

C 是惩罚因子,就是表示模型对误差的宽容度,这个值越高，说明模型对误差越敏感，越不能容忍出现误差。过高会出现过拟合现象，过低会欠拟合。

gamma 是你选择高斯核(RBF)函数作为 kernel 后，该函数自带的一个参数。隐含地决定了数据映射到新的特征空间后的分布，会影响模型的分类精度。

因此想要提高模型效果需要合理的调整参数，本题使用网格搜索法寻找最优参数。

1.4.1.3 网格搜索法

网格搜索法(Grid Search)是一种调参手段，属于穷举搜索，其方法是在所有候选的参数选择中，通过循环遍历，尝试每一种可能性，表现最好的参数就是最终的结果。其流程如下：

首先，对于惩罚系数 C 和 RBF 核参数 gamma 分别选取一个取值范围和步长，从中得到 M 个 C 的值以及 N 个 gamma 值。然后使用穷举法将每一种参数组合构建 M*N 个不同的参数组，使用每个参数构建 SVM 模型得到分类精度，以此确定最优的参数组 C 和 gamma。

最后，如果最佳分类精度所对应的参数值在最初选取的边界，就说明很可能并未选取到最佳参数，需要根据处于那个边界重新选择取值范围，直到最佳参数不在边界为止。此外如果对精度有更高要求可以缩小步长。

网格搜索的优点是可以多参数同时搜索，减少获得最优解所需要的时间。缺点是当参数较多时，计算量巨大。

使用少量数据进行网格搜索法的结果：

```
Parameters:{'gamma': [0.0001, 0.001, 0.01, 0.1, 1, 10, 100], 'C': [0.001, 0.01, 0.1, 1, 10, 100]}
Test set score:0.76
Best parameters:{'C': 10, 'gamma': 0.001}
Best score on train set:0.82
```

图 1.4.1 网格搜索法结果

1.4.1.4 SVM 分类

使用网格搜索法搜索法对部分数据训练得到的参数{'C': 10, 'gamma': 0.001}放入模型进行完整数据的训练。

```
# 调用下SVM模型
clf = SVC(C=10,gamma=0.001,random_state=151) # since we need probabilities
clf.fit(xtrain_svd_scl, ytrain)
predictions = clf.predict(xvalid_svd_scl)
```

训练部分结果如下：

```
array([3, 2, 1, 3, 1, 5, 5, 1, 4, 0, 6, 2, 4, 5, 0, 6, 4, 1, 5, 1, 5, 4,
       5, 2, 1, 5, 1, 4, 4, 4, 5, 2, 4, 5, 4, 4, 5, 5, 4, 0, 1, 5, 2, 4,
       5, 4, 3, 3, 1, 4, 0, 6, 1, 6, 4, 5, 5, 4, 4, 5, 3, 2, 6, 6, 4, 6,
       2, 1, 2, 0, 1, 2, 5, 3, 5, 2, 1, 4, 0, 5, 5, 0, 3, 5, 6, 1, 1, 1,
       4, 1, 4, 5, 1, 1, 0, 4, 0, 3, 1, 5, 1, 1, 5, 6, 4, 1, 4, 4, 6, 5,
       4, 1, 1, 5, 4, 4, 5, 5, 4, 6, 4, 4, 0, 1, 1, 3, 4, 6, 3, 4, 5, 3,
       2, 4, 4, 6, 0, 3, 1, 2, 4, 1, 3, 6, 1, 2, 1, 1, 4, 3, 4, 4, 2, 6,
       4, 2, 3, 4, 4, 1, 6, 0, 3, 4, 3, 1, 6, 5, 4, 4, 5, 1, 2, 2, 4, 3,
       4, 5, 3, 5, 1, 0, 3, 0, 1, 2, 1, 5, 4, 5, 4, 4, 3, 4, 3, 2, 4, 4,
       6, 3, 5, 3, 4, 6, 3, 1, 0, 1, 1, 4, 3, 4, 2, 1, 6, 5, 4, 2, 2, 1,
```

图 1.4.2 SVM 训练结果

其中数字代表标签所映射的自然数。

1.5 分类模型评价

在模型训练前已经将文档一 9:1 的比例分成了训练集和验证集，在使用训练集训练数据后需要将验证集带模型计算验证集的分类结果。为了评价模型的好坏需要引入评价指标。本题使用了 F1-Scores 进行评价：

1.5.1 评价指标介绍

首先介绍在多分类情况下的查准率(Precision)和召回率(Recall)的概念：

$$\text{第 } i \text{ 类的查准率:} \mathrm{P_i} = \frac{TP(i)}{FP(i)+TP(i)} \times 100\%$$

$$\text{第 i 类的召回率:} \mathrm{R_i} = \frac{TP(i)}{TP(i)+FN(i)} \times 100\%$$

其中 $\mathrm{TP}(i)$表示第 i 类中判断正确的样本数量；$FP(i)$表示被误判为第 i 类样本的数量；$FN(i)$表示第 i 类样本被误判为其他类别的数量。[6]

所以查准率可以理解为分类为 i 的样本中有多少是分类正确的，而召回率就是真正的 i 类样本中有多少是被分类正确的。

1.5.2 F1-Scores 介绍

单独考虑 Precision 和 Recall 很难断言一个模型的好坏，这时需要引入 F1-Scores 指标。F1-Scores 是 Precision 和 Recall 的调和平均数综合考虑了两者的性能度量。其表达式为：

$$\mathrm{F1}=\frac{2\times P\times R}{P+R}$$

其中 P 为查准率 R 为召回率，对于多分类有两种计算 F1-Scores 的方法分别是 macro-F1 和 micro-F1

1.5.2.1 macro-F1(宏 F1)

将多分类任务的两两类别的组合都对应一个混淆矩阵，其 Precision 和 Recall 的组合记为$(\mathrm{P}_1,\mathrm{R}_1),(\mathrm{P}_2,\mathrm{R}_2),\ldots,(\mathrm{P}_n,\mathrm{R}_n)$再分别计算 P 和 R 的平均值得到 macro-P 和 macro-R：

$$\text{macro-P(宏查准率)}=\frac{1}{\mathrm{n}}\sum_{i=1}^{n}P_i$$

$$\text{macro-R(宏查全率)}=\frac{1}{\mathrm{n}}\sum_{i=1}^{n}R_i$$

$$macro\text{-}F1\text{(宏 F1)}=\frac{2\times macro\text{-}P\times macro\text{-}R}{macro\text{-}P+macro\text{-}R}$$

1.5.2.2 micro-F1(微 F1)

将所有混淆矩阵所对应的元素进行平均，得 TP、FP、TN、FN 的平均值，分别记为$\overline{\mathrm{TP}}$、$\overline{\mathrm{FP}}$、$\overline{\mathrm{TN}}$、$\overline{\mathrm{FN}}$，再计算得到：

$$\text{micro-P} = \frac{\overline{TP}}{\overline{FP} + \overline{TP}} \times 100\%$$

$$\text{micro-R} = \frac{\overline{TP}}{\overline{TP} + \overline{FN}} \times 100\%$$

$$\text{micro-F1} = \frac{2 \times \text{micro}-P \times \text{micro}-R}{\text{micro}-P + \text{micro}-R}$$

1.5.3 F1-Scores 的调用与结果

在 sklearn 中调用 f1_score 函数并将参数设为 micro-F1

```
print ("模型的f1_score: %0.3f " % f1_score(yvalid, predictions, average='micro'))
模型的f1_score: 0.914
```

图 1.5 F1-Scores 评价结果

1.6 结果展示

	留言编号	留言用户	留言主题	留言时间	留言详情	一级标签	模型结果
8030	154061	U0007047	茶叶生产许可证能否简政放权	2018/6/14 17:32:20	\n\t\t\t\t\t\n\t\t\t\t\t茶叶是富民产业，也是易种易学产业，打造千亿茶...	商贸旅游	商贸旅游
9115	303508	U0004369	E12市卫计局重复征收社会抚养费	2019/2/1 10:53:47	\n \n 我叫曹正科，电话**********.我2017年...	卫生计生	卫生计生
6698	276283	U0004465	早年间参加工作的企业退休工人连续5年没有增调工资了	2018/8/20 13:29:08	\n \n 尊敬的领导：今年喜闻西地省退休人员普调工资，感谢政府对退休人员的...	劳动和社会保障	劳动和社会保障
7181	17860	U0001630	A市星沙镇母亲被骗进传销，我该怎么办?	2016/11/11 14:56:14	\n\t\t\t\t\t\n\t\t\t\t\t 母亲在今年七月份被骗去在西地省A市星沙镇...	商贸旅游	商贸旅游
6073	139113	U0001444	请I市政府帮助受工伤农民工获得应有赔偿	2015/12/14 11:10:32	\n \n 尊敬的市长： 我叫祝落根，男，1970年出生，I市I1区此湖口...	劳动和社会保障	劳动和社会保障
5003	325117	U0006961	G7县教育局中考录取存在暗箱操作	2019/7/1 18:08:42	\n \n 1、一中录取学生分数不阳光公布，存在低分录取的嫌疑2、同城的公立...	教育文体	教育文体
4141	118072	U0004187	C3县教育局人治独有情钟、恋恋不舍	2015/5/1 10:48:03	\n \n C3县教育局为什么对人治独有情钟和恋恋不舍?因为昔日人治何等...	教育文体	教育文体
5619	72172	U0004278	F3区柳林镇农机站老人十年艰苦讨薪路	2013/12/24 11:41:34	\n \n 尊敬的F3区委书记： 您好！我爸爸付名儒今年7...	劳动和社会保障	劳动和社会保障
832	91604	U0005453	揭露J2区南塔办事处违规拆迁改造行为	2013/6/22 12:42:50	\n\t\t\t\t\t\n\t\t\t\t\t J2区南塔办事处在原住宅建设公司现居民六...	城乡建设	城乡建设
3313	166794	U0006120	F市民价值上万的货物在物流仓库被弄丢，我该找哪个部门维权?	2019/7/4 12:07:15	\n\t\t\t\t\t\n\t\t\t\t\t您好！去年11月14日我38件实木地板，价值...	交通运输	交通运输

图 1.6 最终分类结果

其中模型结果列为最终一级标签分类模型的分类结果

2 热点问题挖掘

2.1 流程分析

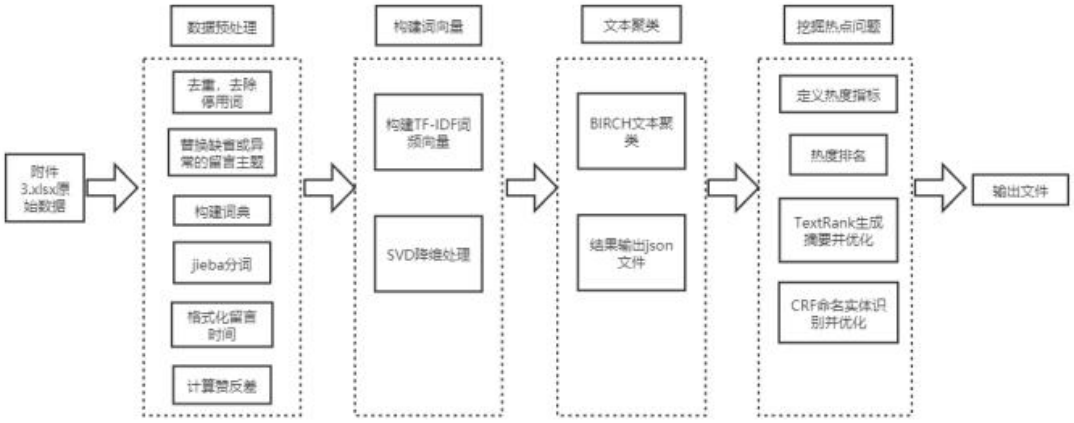


图 2.1 总体流程图

本题使用的编程语言为 Python3，采用聚类算法对留言数据进行归类，把描述相同的问题聚为一类，再根据热度算法对留言进行排序，最后生成热点问题的摘要和提取问题中的对象(地点/人群)。

2.2 数据预处理

2.2.1 构建用户词典

经过试验，我们发现 jieba 分词库对地区名的识别不够准确，因此需要构建一个用户词典让其对地区名和交通线路名称识别更加精准。我们根据正则匹配算法匹配出留言数据来源于“西地省”，数据中行政地名皆用大写字母和数字混合

组成，如“A 市”、“A3 区”、“B2 县”等。根据数据中的这些命名规律，我们用 Python3 对 26 个字母和 1～20 的数字生成了行政地区名；用数字 n (n ∈ {,1,2,…,999})生成“n 路公交车”的公交线路名称；用数字 m (m ∈ {,1,2,…,12})生成“地铁 m 号线”的地铁线路名称。对于道路名，我们从网上搜集了全国最常用的数百个道路名称。我们将以上获得的名词添加进用户词典，再使用 jieba 分词库分此后，效果如下：

```
jieba分词前的句子：'A3区一米阳光摄影是否合法纳税了？劳动路有油烟扰民，关于地铁1号线拥堵情况的建议，108路公交车总排班太少了'
加载用户词典前的分词效果：'A3区 一米阳光 摄影 合法 纳税 劳动路 油烟 扰民 地铁1号线 拥堵 情况 建议 108路公交车 总 排班 太 少'
加载用户词典后的分词效果：'A3区 一米阳光 摄影 合法 纳税 劳动路 油烟 扰民 地铁1号线 拥堵 情况 建议 108路公交车 总 排班 太 少'
```

图 2.2.1 分词效果

2.2.2 词性标注和分词

留言主题是留言用户反映问题的总结和归纳，而留言详情描述该问题的内容要素繁多且复杂，相比较下留言主题更能体现出这条留言的特征和含义，故本题选用留言主题的数据进行文本聚类。此步骤对留言数据进行预处理：

1. 对留言数据去重，去除停用词
2. 使用 HanLP 从留言详情中抽取摘要，替换缺省或异常的留言主题内容
3. 构建用户词典
4. 用 jieba 加载用户词典对留言主题进行词性标注和分词
5. 格式化处理留言时间
6. 计算每一条留言的“点赞数”减“反对数”的值

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	主题分词	主题分词_词性	点赞反对差
188006	A00010294	A3区一米	2019/2/28	座落在A市	0	0	A3区 一米阳光 婚纱 艺术	A3区/ns 一米阳光/nz 婚纱/n 艺术摄影/	0
188007	A00074795	咨询A6区	2019/2/14	A市A6区道	0	1	A6区 道路 命名 规划 初步	A6区/ns 道路/n 命名/n 规划/n 初步/d	1
188031	A00040066	反映A7县	2019/7/19	本人系春华	0	1	A7县 春华 镇 金鼎村 水泥	A7县/ns 春华/nz 镇/n 金鼎村/nr 水泥	1
188039	A00081379	A2区黄兴	2019/8/19	靠近黄兴	0	1	A2区 黄兴路 步行街 古道	A2区/ns 黄兴路/nr 步行街/n 古道/n 巷	1
188059	A00028571	A市A3区中	2019/11/2	A市A3区中	0	0	A市 A3区 中海 国际 社区	A市/ns A3区/ns 中海/ns 国际/n 社区/n	0
188073	A909164	A3区麓泉	2019/3/11	作为麓泉	0	0	A3区 麓 泉 社区 单方面	A3区/ns 麓/ng 泉/ns 社区/n 单方面/n	0
188074	A909092	A2区富绿	2019/1/31	“二高一	0	0	A2区 富绿 新村 房产 性质	A2区/ns 富绿/nr 新村/ns 房产/j 性质/	0
188119	A00035029	对A市地铁	2019/5/27	我是一名	0	0	A市 地铁 违规 用工 质疑	A市/ns 地铁/n 违规/vn 用工/n 质疑/v	0

图 2.2.2 预处理后的数据

2.3 构建词向量

使用 TF-IDF 方法构建留言主题的词向量，再使用 SVD 奇异值分解方法对词向量进行降维处理，详细方法参见上文 1.3 文本空间向量模型，在此不再赘述。

2.4 基于 BIRCH 的文本聚类

2.4.1 BIRCH 算法原理

本题中的文本数据量大、没有标注，且不确定类数，在现有条件下无法使用人工标注的方法构建训练样本，因此对文本进行聚类挖掘。文本聚类为本题热点问题挖掘的重要部分，它依据的是文本之间的相似度，将文本集合自动归类，并尽可能使内容相似度较大的文本划分为同一类。

聚类就是给定一个包含 N 个数据点的数据集和一个距离度量函数 F(例如计算簇内每两个数据点之间的平均距离的函数)，要求将这个数据集划分为 K 个簇(或者不给出数量 K，由算法自动发现最佳的簇数量)，最后的结果是找到一种对于数据集的最佳划分，使得距离度量函数 F 的值最小。从机器学习的角度来看，聚类是一种非监督的学习算法，通过将数据集聚成 n 个簇，使得簇内点之间距离最小化，簇之间的距离最大化。

BIRCH 算法属于层次聚类，即层次方法来聚类和规约数据，适合于数据量大，类别数 K 比较多且不确定的情况。因此本题采用 BIRCH 聚类算法对“留言主题”进行聚类。BIRCH 算法中引入了两个概念：聚类特征和聚类特征树，以下分别介绍。

2.4.1.1 聚类特征(CF)与聚类特征树(CF Tree)

CF 特征是 BIRCH 聚类算法的核心，一个 CF 即为一个三元组，它表示簇的所有信息，可以用(N, LS, SS)表示，其中 N 表示此 CF 中所含的样本点数量，LS 表示此 CF 中所含的样本点各特征维度的和向量，SS 表示此 CF 中所含的样本点各特征维度的平方和：

若在 CF 中有 5 个样本(3,4)，(2,6)，(4,5)，(4,7)，(3,8)，则此 CF 对应的：

$$N = 5,$$

$$LS = (3 + 2 + 4 + 4 + 3, 4 + 6 + 5 + 7 + 8) = (16, 30),$$

$$SS = (32 + 22 + 42 + 42 + 32 + 42 + 62 + 52 + 72 + 82) = 244$$

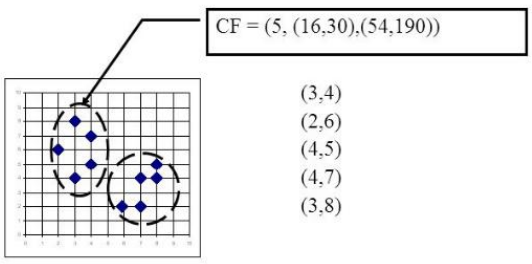


图 2.4.1 CF 示意图

CF Tree 的结构类似于一棵 B-树，它有两个参数：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 $(CF_i,Child_i),1 \le i \le B$，$CF_i$是这个节点中的第 i 个聚类特征，$Child_i$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[7]。如图所示：

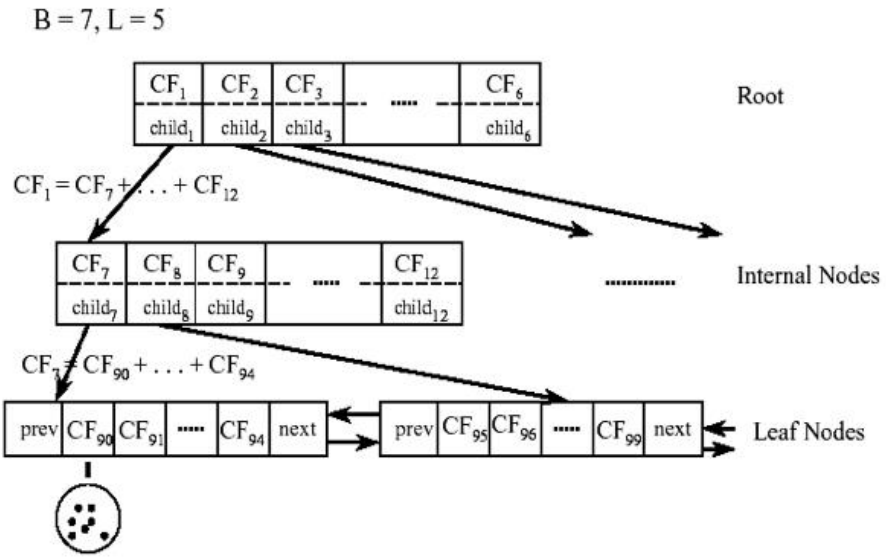


图 2.4.2　CF Tree 示意图

CF 有可以求和的特性，也就是满足线性关系：

把这个性质延伸到 CF Tree 上，对于每个父节点中的 CF 节点，它的(N, LS,SS)三元组的值等于这个 CF 节点所指向的所有子节点的三元组之和:

$$CF_1 + CF_2 + \ldots + CF_i$$
$$= (n_1 + n_2 + \ldots + n_i, LS_1 + LS_2 + \ldots + LS_i, SS_1 + SS_2 + \ldots + SS_i)$$

于是引出簇的质心和簇的半径的概念。假如一个簇中包含 n 个数据点：$\{X_i\}$，$i = 1,2,3 \ldots n$，则质心 C 和半径 R 计算公式如下：

$$C = \frac{X_1 + X_2 + \ldots + X_n}{n}$$

$$R = \frac{|X_1 - C|^2 + |X_2 - C|^2 + \ldots + |X_n - C|^2}{n}$$

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，当我们把一个数据点加入某个簇的时候，此数据点的详细特征，例如属性值将会丢失，对于此特性，BIRCH 聚类算法可以在很大程度上对数据集进行压缩。

2.4.1.2 生成聚类特征树 CF Tree

首先定义好 CF Tree 的参数：B 为内部节点最大 CF 数， L 为叶子节点的最大 CF 数， T 为叶节点中的每个 CF 最大样本的半径阈值。在初始状态，没有任何样本，CF Tree 为空，然后从训练集读入第一个样本点，将它放入一个新的 CF 三元组 A，这个三元组的 N=1，将这个新的 CF 放入根节点，此时的 CF Tree 如下图：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。随着数据点的加入，CF 树被动态的构建，插入过程类似于 B-树。加入算法表示如下：

(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。(如图 2.4.3-a)

(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)。(如图 2.4.3-a)

(3) 如果CF节点当前的叶节点的数量小于阈值L,然后创建一个新的CF节点,一个新的样品,一个新的CF节点放入这个叶子节点,路径上的所有CF三元组更新,插入结束。否则，转到步骤(4)。(如图 2.4.3-b)

(4) 将当前的叶子节点划分为两个新的叶子节点，在旧叶子节点的所有 CF 元组中选择超球面距离最长的两个 CF 元组，作为两个新叶子节点的第一个 CF 节点进行分配。根据距离原理将其他元组和新的样本元组放入相应的叶节点中。检查父节点是否也被分割，如果需要的话，以与叶节点相同的方式进行分割。(如图 2.4.3-c)[8]

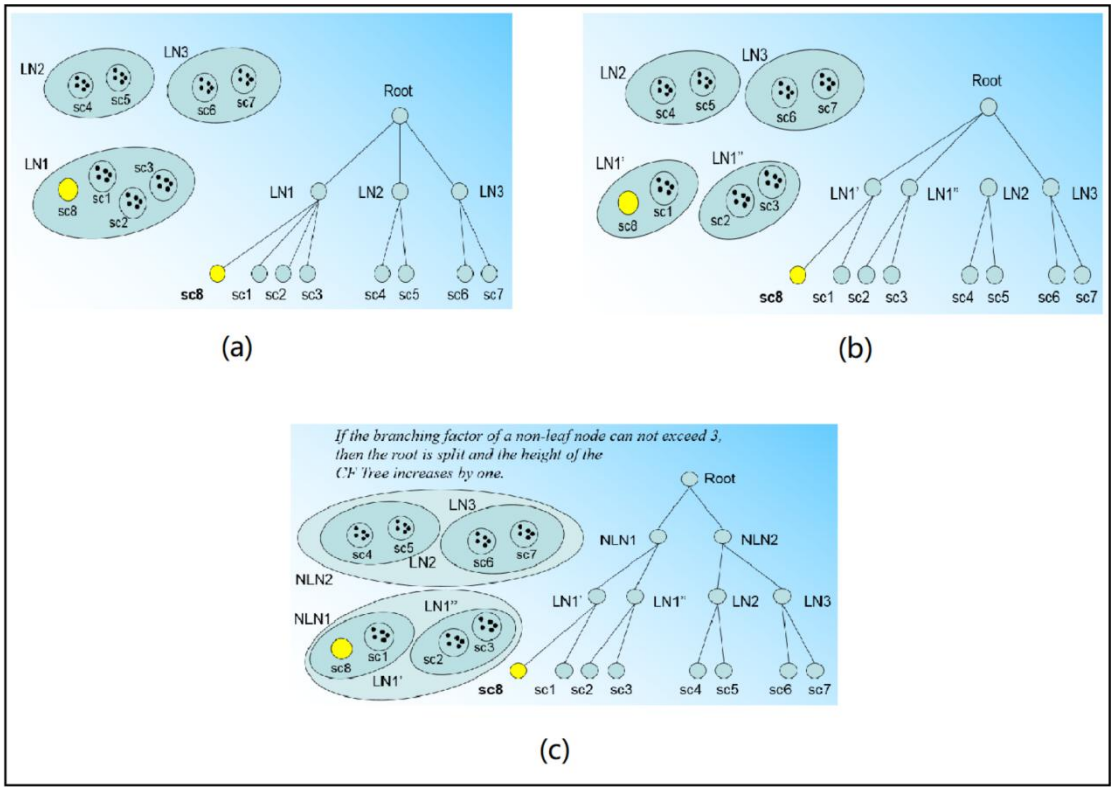


图 2.4.3 生成 CF Tree

2.4.1.3 BIRCH 算法

将所有的训练集样本建立了 CF Tree，一个基本的 BIRCH 算法就完成了，对应的输出就是若干个 CF 节点，每个节点里的样本点就是一个聚类的簇。也就是说建立 CF Tree 的过程就是 BIRCH 算法的主要过程。

2.4.1.4 BIRCH 原理小结

BIRCH 算法可以不用输入类别数 K 值，如果不输入 K 值，则最后的 CF 元组的组数即为最终的 K，否则会按照输入的 K 值对 CF 元组按距离大小进行合并。BIRCH 算法适用于样本量较大的情况。

2.4.2 BIRCH 文本聚类

本题调用 python3 中的 sklearn.cluster.Birch 对留言主题的词向量进行 BIRCH 聚类，发现 branching_factor=160，threshold=0.82 时取得了最好效果，如图 2.4.4：

图 2.4.4 BIRCH 文本聚类效果日志

可以看出经过 BIRCH 聚类后，某一时段内反映特定地点或特定人群问题的留言被归类，并计算出了每一类问题的留言个数。我们将聚类后的每一条留言数据按照聚类标签保存到 json 文件中(如图 2.4.5)，格式为此类问题的留言数、留言主题用中文句号拼接而成的字符串、留言编号列表、留言时间的范围、最新的留言时间戳以及点赞数减去反对数的值，以便下一步进行热点问题的挖掘。


```
{
  "留言数": 2,
  "留言主题": "A6区月亮岛路架设高压电线环评造假民众做主。反对A6区月亮岛路架设高压电线强烈要求重启环境评估。",
  "留言编号": [
    218442,
    231773
  ],
  "时间范围": "2019-04-08 21:19:40 至 2019-04-12 14:59:14",
  "最新时间戳": 1555052354.0,
  "点赞数-反对数": 23
},
```


图 2.4.5 聚类结果保存在 json 文件中的格式

2.5 挖掘热点问题

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”。建立合理的热度指标能及时发现热点问题，有助于相关部门进行有针对性地处理，提升服务效率。

2.5.1 热度指标

2.5.1.1 定义热度指标

热度指标是评价某类问题是否能成为热点问题的衡量标准。基于题目给出的数据和 BIRCH 聚类后的结果，我们发现其中留言时间、点赞数、反对数和每一类的留言数可作为热度评价指标的因变量。

我们把点赞数减去反对数得到真正的用户点赞支持数，表示为赞反差 x，并将 x 表示为 z；把此类问题的留言数表示为 s；把此类问题中最新的留言时间距离 2012 年 12 月 25 日 00:00:00 的天数表示为 t，热度指数表示为 Score，则有：

$$z = \begin{cases} 1, & x < 1 \\ x, & x \geq 1 \end{cases}$$

$$Score = \frac{s \times t}{86400} \times (1 + 0.1 \times \log_2 z)$$

此算式中，x、t 和 s 均与热度指数 Score 成正相关(如图 2.5.1)，即：

(1) 赞反差越大，热度指数越高。

(2) 留言数越多，热度指数越高。

(3) 时间越新，热度指数越高，并且成线性增长。

(4) 与赞反差相比，热度指数对留言数更为敏感。

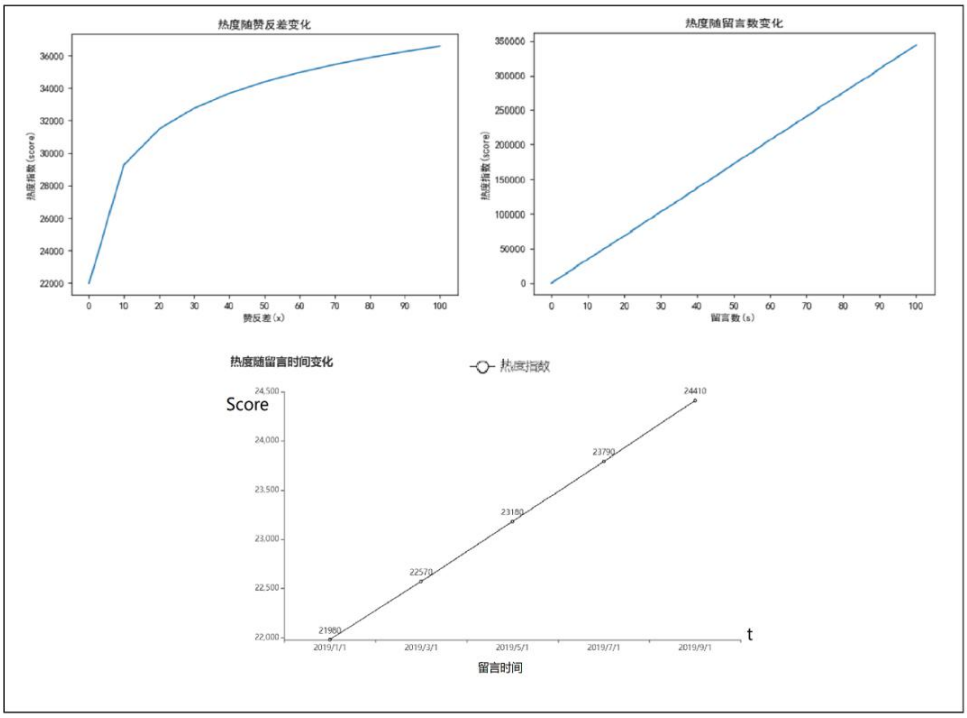


图 2.5.1 热度指数与因变量的关系

2.5.1.2 计算热度指数

定义好热度指标后，我们读取聚类后生成的 json 文件，对每一类留言进行热度指数的计算，然后将这些留言问题按照的热度指数降序来排序，最终得到热度指数排名前 5 的留言问题。(如图 2.5.2)

A	B	C	D	E	F	G	H
热度排名	问题ID	热度指数	时间范围	其	问	问题id	
1	1	269201.74	2019-04-10 10:21:47 至 2020-01-26 19:47:11	A	A	[264806,	193091, 254
2	2	160731.57	2019-07-07 07:28:06 至 2019-12-31 21:00:00	A	A	[289950,	283879, 223
3	3	104302.62	2018-11-15 16:07:12 至 2019-12-02 11:57:49	A	A	[289408,	224042, 282
4	4	96024.763	2019-03-05 10:24:30 至 2020-01-07 13:15:05	A	A	[248415,	280000, 248
5	5	92336.153	2019-01-07 11:37:23 至 2019-12-11 23:24:48	A	A	[243062,	224045, 206

图 2.5.2 热度指数前五排名结果

2.5.2 文本摘要

在上一节中我们已经完成了对热点问题的排序，本节中我们生成热点问题的问题描述。在聚类后生成的 json 文件中，留言主题的数据是以中文句号拼接每条留言主题而成的一个字符串(如图 2.4.5)，因此当留言数大于 1 时，我们需要对留多条留言主题拼接成的字符串进行摘要生成，以提取出该类问题的准确问题描述。在本题中，我们使用 TextRank 算法生成摘要作为问题描述。如果留言数只有一条，则直接把该条留言的留言主题作为问题描述。

2.5.2.1 TextRank 原理

TextRank 算法是一种基于 Google 的 PageRank (Brin and Page, 1998)算法，属于基于图(Graph-based)的模型，它通过词之间的相邻关系构建拓扑结构，然后使用 PageRank 算法对每个节点的 rank 值进行排序，即可根据 rank 值得到关键词。PageRank 本来是用来解决网页排名的问题，网页即为结点，网页间的关系为图的边。PageRank 的公式如下：

$$S(V_i) = (1-d) + d * \sum_{j \in In(V_i)} \frac{1}{|Out(V_j)|} S(V_j)$$

其中，$S(V_i)$表示结点V_i的 rank 值，$In(V_i)$表示结点V_i的前驱结点集合，$Out(V_j)$表示结点V_j的后继结点集合，d 为阻尼系数(damping factor)，一般取 0.85。

TextRank 将这种思想运用到句子处理中，将词当成结点，某一个词与其前面的 N 个词、以及后面的 N 个词均具有图相邻关系，而词与词之间的关系就是图的边。

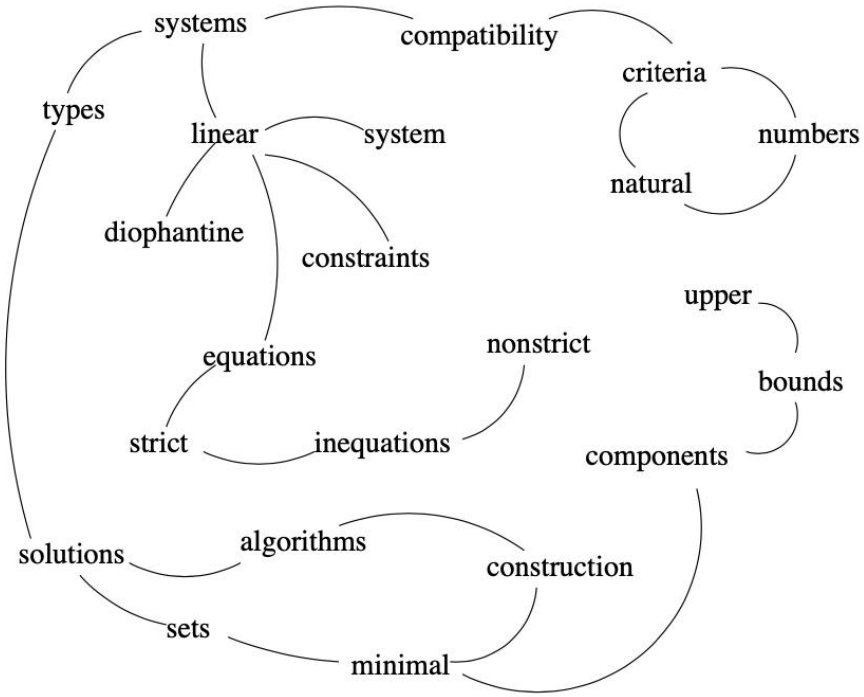


图 2.5.3　具有图相邻关系的词

根据 TextRank 的定义，将 PageRank 的公式改为：

$$\mathrm{WS(V_i)} = (1-\mathrm{d}) + \mathrm{d} * \sum_{j \in In(V_i)} \frac{1}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

可以看出，该公式仅仅比 PageRank 多了一个权重项W_{ji}，用来表示两个节点之间的边连接有不同的重要程度。

2.5.2.2 TextRank 用于摘要提取的流程

(1) 把所有文章整合成文本数据，并把文本分割成单个句子，即:

$$T = [S_1, S_2, \ldots S_m] \#(1)$$

(2) 对于每个句子属于 T，进行分词和词性标注处理，并过滤掉停用词，只保留指定词性的单词，如名词、动词、形容词，即:

$$S_i = [t_{i,1}, t_{i,2}, \ldots, t_{i,n}] \#(2)$$

其中$t_{i,j}$是保留后的候选关键词。

(3) 将每个句子中所有单词的词向量合并为句子的向量表示。

(4) 计算句子向量间的相似性并存放在矩阵中，作为转移概率矩阵 M。然后将矩阵 M 转换为以句子为节点、相似性得分为边的图结构，用于句子 TextRank 计算。

(5) 根据上面公式，迭代计算各节点的权重。

(6) 对节点权重进行倒序排序，从而得到最重要的 T 个句子，作为候选摘要。

(7) 将上一步得到的最重要的 T 个句子，在原始文本 T 中进行标记，若形成相邻句，则组合成多句摘要。

2.5.2.3 TextRank 摘要

HanLP 自然语言处理工具包中的 extractSummary 函数实现了基于 TextRank 算法的摘要功能，我们调用 extractSummary 函数后的效果如下：

```
摘要前：'西地省A市2017年出租车油补发放。A市出租车燃油补贴发放。A市A2区出租车2017年燃油补贴发放。请求落实2017年2018年A市区教职工单位文明奖发放。A市2018年度出租车燃油补贴发。'
摘要后：'A市出租车燃油补贴发放
```

图 2.5.4 TextRank 摘要结果

2.5.2.4 摘要效果优化(地名提取算法)

从图 2.12 的效果可以看出，对于多条留言主题组成的字符串进行在摘要生成，取得了较好的效果，但是对于地名的概括性还不够精确。如摘要前的留言中，我们可以看出问题所反映的完整地区是“西地省 A 市 A2 区”，而摘要后的结果是“A 市”，摘要算法是以一个句子为单位进行提取的，所以并不能对每个句子内的词语进行抽取。对此，我们设计了一个行政地名提取的算法，流程如下：

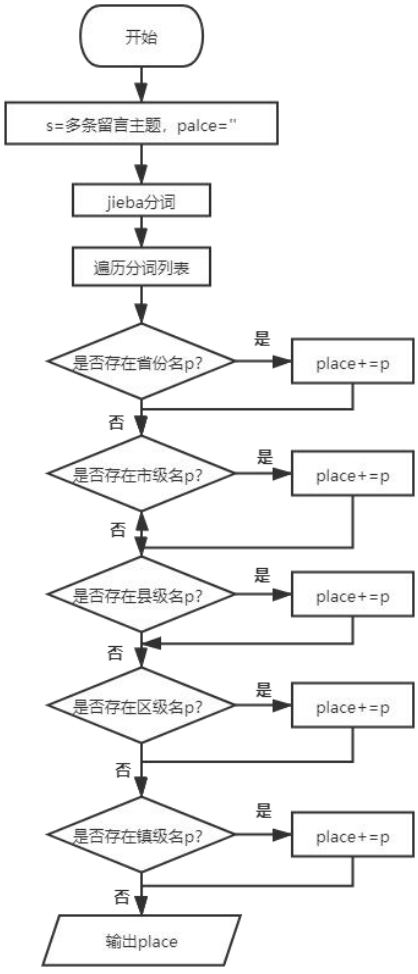


图 2.5.5 地名提取算法

使用地名提取算法后的效果如下：

摘要前：'西地省A市2017年出租车油补发放。A市出租车燃油补贴发放。A市A2区出租车2017年燃油补贴发放。请求落实2017年2018年A市区教职工单位文明奖发放。A市2018年度出租车燃油补贴发。'
摘要后：'A市出租车燃油补贴发放
地名提取算法摘要后：'西地省A市A2区出租车燃油补贴发放'

图 2.5.6 地名提取算法效果

2.5.3 命名实体识别

2.5.3.1 命名实体识别简介

对完成对留言问题的描述后，根据题意还需要识别出留言问题中的主体对象，如问题中的某个地点或某类人群。这就需要用到命名实体识别(NER)。

NER(Named Entity Recognition)又称作专名识别，是自然语言处理中的一项基础任务，应用范围非常广泛。命名实体一般指的是文本中具有特定意义或者指代性强的实体，通常包括人名、地名、组织机构名、日期时间、专有名词等，目前主流的做法是把命名实体识别转换为一个序列标注的问题。在本题中，只需要识别出人名、地名、组织机构名即可。

2.5.3.2 基于 CRF 模型的 NER

条件随机场(Conditional Random Fields)，简称 CRF，是一种判别式的概率图模型。条件随机场是在给定随机变量 X 条件下，随机变量 Y 的马尔可夫随机场。原则上，条件随机场的图模型布局是可以任意给定的，但比较常用的是定义在线性链上的特殊的条件随机场，称为线性链条件随机场。CRF 是序列标注场景中常用的模型，比基于隐马尔可夫模型(HMM)的最短路径分词能利用更多的特征，比 MEMM 更能抵抗标记偏置的问题。

本题使用HanLP实现的基于CRF模型中文分词器来对问题描述进行命名实体识别，得到的结果如下：

```
待识别文本：'A7县泉塘街道楚役家园停电'
CRF命名实体识别结果：'[A7县/ns 泉塘街道/ns 楚役/nz 家园/n]/nt 停电/v'
```

2.5.6 CRF 命名实体识别结果

根据 HanLP 的词性标注解释，词性“nt”表示地名，从识别结果可以看出，“A7 县泉塘街道楚役家园”被成功识别为地名实体。我们把机构组织相关(医院、学校、公司等)以及地名和其他专名的词性，标记为作为实体识别结果输出。

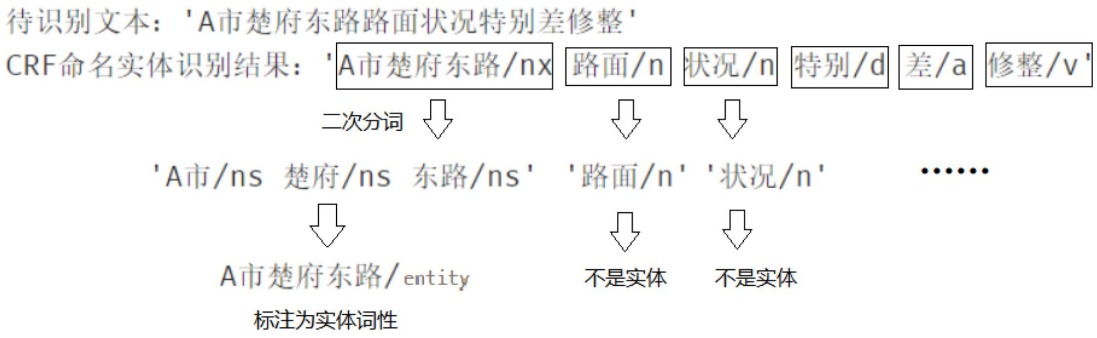
2.5.3.3 命名实体识别优化

在 CRF 命名实体的识别中，有时存在实体识别不出来的情况：

```
待识别文本：'A市楚府东路路面状况特别差修整'
CRF命名实体识别结果：'A市楚府东路/nx 路面/n 状况/n 特别/d 差/a 修整/v'
```

2.5.7 CRF 命名实体识别结果误差

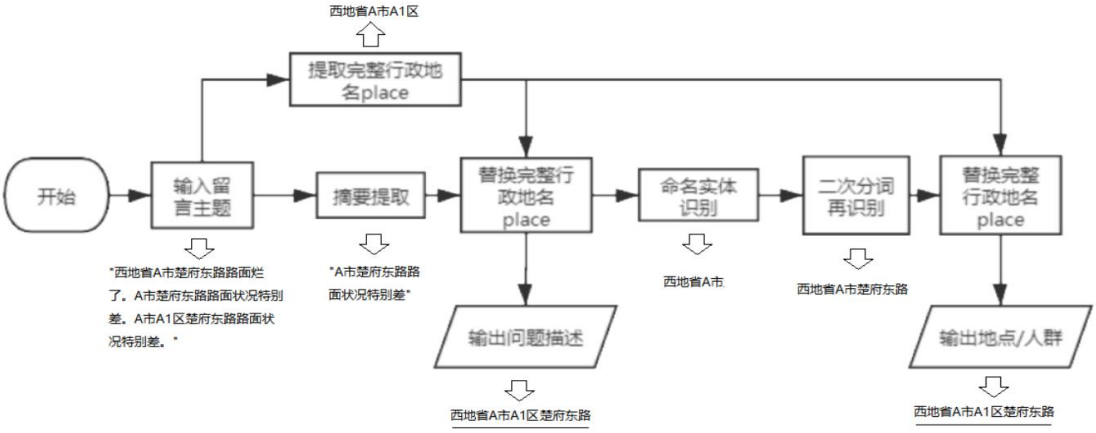
如图 2.5.7，“A 市楚府东路”被标注为“nx”字母专名词性，这显然是不符合实际的。为了纠正这个误差，我们对 CRF 分词后的词性列表中，没有含有实体词性的分词结果进行二次分词处理，若二次分词后的词列表中有属于实体词性的词，则把这个二次分词前的词标注为实体词性“entity”：



2.5.8　二次分词纠正误差

对于二次分词后仍然没有识别出实体词性的句子，我们将原句作为一个实体输出。

由于 HanLP 的 CRF 模型中文分词器不支持用户词典，所以也存在某个地名无法完整识别出来的情况。对此我们从 2.5.2.4 摘要效果优化(地名提取算法)中获取完整行政地名，用完整地名替换掉实体识别结果中含有的不完整行政地名，如图 2.5.9：



2.5.9　优化后流程图

2.5.4 输出结果表

我们根据 BIRCH 聚类的结果，先计算热度指数并选出最高的 5 个热点问题，再对热点问题进行 TextRank 摘要提取出问题描述，并对其进行优化，然后从摘要中提取出命名实体“地点/人群”，并对结果进行优化，按照题目所给的格式输出为两个文件，如图 2.5.10、图 2.5.11：

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	269201.735	2019-04-10 10:21:47 至 2020-01-26 19:47:11	A2区新城搅拌站	A2区丽发新城建搅拌站噪音扰民
2	2	160731.574	2019-07-07 07:28:06 至 2019-12-31 21:00:00	A市伊景园滨河苑	A市伊景园滨河苑捆绑车位销售
3	3	104302.623	2018-11-15 16:07:12 至 2019-12-02 11:57:49	A市人才补贴	A市人才补贴
4	4	96024.763	2019-03-05 10:24:30 至 2020-01-07 13:15:05	A7县泉塘街道	A7县泉塘街道成菜园
5	5	92336.153	2019-01-07 11:37:23 至 2019-12-11 23:24:48	A7县楚龙街道	A7县楚龙街道拆迁

2.5.10 热点问题表

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	264806	A000107694	A市丽发新城旁商品交	2019/4/10 10:21	丽发新城旁C5市南路上摆了	0	3
1	193091	A00097965	A市富绿物业丽发新城	2019/6/19 23:28	位于A市A2区暮云街道丽发新	0	242
1	254710	A00092242	A市丽发新城2期和3期	2019/6/25 16:30	尊敬的领导：您好！我是A市	0	1
1	191943	A00038563	A市A2区丽发新城道路	2019/7/3 12:03	A市A2区丽发新城第一期与第	0	1
1	219174	A00081998	A2区丽发新城小区内与	2019/7/3 23:27	A2区丽发新城二期45栋，在	0	3
1	284576	A00063717	A2区丽发新城小区云堵	2019/7/26 17:41	A市A2区暮云街道丽发新城小	0	0
1	265342	A000106707	A市丽发新城三期违建	2019/8/4 17:10	我是最近看到新闻说丽发新	0	5
1	267050	A909227	噪音、灰尘污染的A2区	2019/11/2 10:18	A2区丽发新城附近修建搅拌	0	0
1	264944	A0004260	A2区丽发新城附近修建	2019/11/2 14:23	A市A2区丽发新城小区附近，	0	0
1	189950	A909204	投诉A2区丽发新城附近	2019/11/13 11:20	我是A2区丽发新城小区的一	0	0
1	281943	A909216	举报A2区丽发新城小区	2019/11/15 8:56	A2区辖区内，丽发新城小区	0	0
1	225217	A909223	A2区丽发新城附近修建	2019/11/15 9:17	我已经好久没睡过安稳觉了	0	0
1	243692	A909201	丽发新城小区附近的搅	2019/11/15 11:23	领导您好！我是暮云街道丽	0	2
1	255008	A909208	投诉小区附近搅拌站噪	2019/11/18 12:23	暮云街道丽发新城边上在建	0	0
1	203393	A00053065	A市丽发新城小区侧面	2019/11/19 14:51	发同投资有限公司在未经业	0	2
1	188809	A909139	A市万家丽南路丽发新	2019/11/19 18:07	A市万家丽南路丽发新城居民	0	1

2.5.11 热点问题留言明细表

2.5.12 热点问题词云图

3 答复意见的评价

3.1 总体流程

本题的总体思路如下：

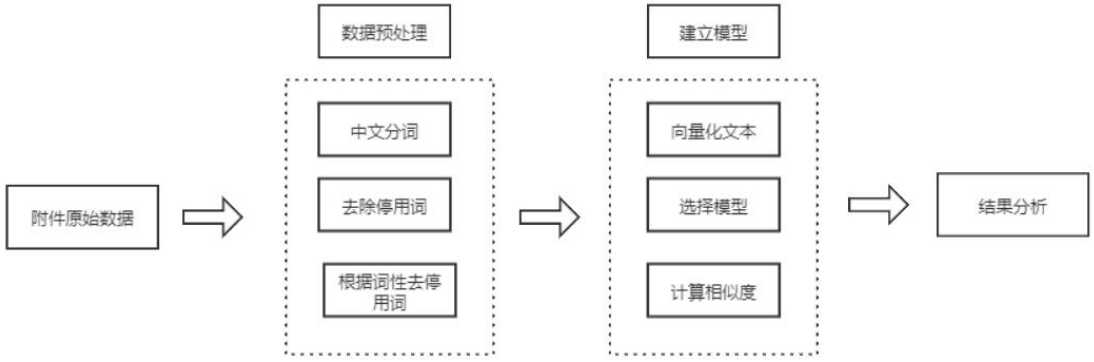


图 3.1.1 计算相关度的流程图

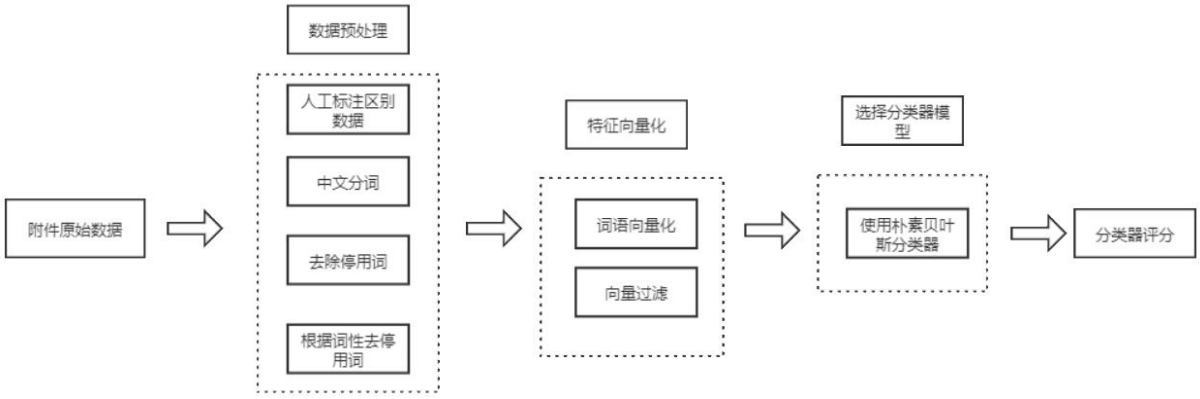


图 3.1.2 计算是否可解释的流程图

3.2 数据预处理

3.2.1 文本预处理

本题需要对赛方所给的文本中做文本预处理。

（一） 符号、停用词过滤

在附件 4 本题所需的文本数据中，因为一些语气词、副词、符号、数字通常本身无明确的意义，并且如果保留不处理会影响本题在后面的模型处理，所以需要借助中文停用词词典将其过滤。

（二） 中文分词

中文分词是中文文本处理的第一个基础的步骤，跟英文句子不同，因为中文句子词与词之间没有相互的界限，所以在进行对中文文本处理时要首先对其进行分词处理。

中文分词基于其实现原理跟特点可以分为两种：

(1) 基于词典分词算法

词典分词是按照一定策略将要匹配的字符串跟一个准备好的大型词典的词进行匹配。若找到某个词条，则被判定为匹配成功，识别此词。此类算法一般会加入一些“正向\反向最大匹配”等策略。此类算法优点是速度快，因为它们的时间处理复杂度都是 O(n),比较简单。缺点是对于歧义词与未登录词处理结果并不理想。

(2) 基于机器学习的算法

此类算法基本思路是对汉字进行标注训练，考虑词语出现的频率同时也要考虑对上下

的影响，具有比较好的学习能力但是需要大量的人工标注数据。对歧义词和未登录词都有比较好的效果。

本题采用 python 中的一个分词库 jieba(结巴分词)对附件 4 中的留言文本和答复文本进行处理。涉及的的算法上文已给出

使用 jieba 分词部分结果如下：

```
0       [位于, 市, A2, 区, 桂花, 坪, 街道, A2, 区, 公安分局, 宿舍区, 景蓉...
1       [潇楚, 南路, 修, 快, 路, 挖, 稀烂, 围栏, 围, 不怎么, 动工, 有时候, ...
2       [地处, 省会, 市, 民营, 幼儿园, 小孩, 祖国, 民营, 幼儿园, 教师, 超负荷,...
3       [尊敬, 书记, 您好, 研究生, 毕业, 人才, 新政, 落户, 市, 想, 买, 套, ...
4       [建议, 白竹坡, 路口, 更名, 马坡岭, 小学, 原, 马坡岭, 小学, 取消, 保留,...
                              ...
2811    [市, 汽车, 北站, 进站, 口, 居民, 马良, 社区, 马园口, 组, 新, 大楼, ...
2812    [强烈, 市, 路, 公交车, 改, 线路, 获悉, 路, 公交车, 更改, 线路, 滨江路...
2813    [G7, 县文, 盛, 小学, 引入, 特色, 班, 学生, 参加, 特色, 班, 老师, ...
2814    [贺, 厅长, 燃油, 税费, 改革, 市, 财政局, 咨询, 得不到, 现, 咨询, 中央...
2815    [A8, 县, 朱良桥, 乡, 说, A8, 县, 破烂, 乡, 集镇, 建设, 相关, 基...
Name: 留言详情_Length: 2816, dtype: object
```

图 3.2.1 jieba 分词部分结果

（三） 词性过滤

因为过多的副词量词并不具备特征，会在后面影响模型的训练，停用词词典也不能去除所有的这些无特征的词语，而附件 4 所读取的[留言详情]文本会附带大量的转义字符，所以这里本题还需对分词后的词性再次做一部分过滤。对['x', 'c', 'u','d', 'p', 't', 'uj', 'm', 'f', 'r']等词性做一次过滤。例如：

'\n\t\t\t\t\t\n\t\t\t\t\t2019 年 4 月以来，位于 A 市 A2 区桂花坪街道的 A2 区公安分局宿舍区(景蓉华苑)出现了一番乱象 ..... 面对公安干警采用这种方式投票合法性在哪？\n\t\t\t\t\t\n\t\t\t\t\t\n\t\t\t\t\t'

（四） 人工标注信息

在判别附件 4[答复意见]时候,本题对其进行了人工标注，例如解释性差的标注为 0，解释下好的标注为 1。

区景蓉华苑物业管理有问兖计，超过三分之二的业主同意收取停车管理费，在业主大会结束后业委会也对业主提出的意见和建议进	2019/5/10 14:56:53	1
蕖楚南路洋湖段怎么还没怵标准高，该段原路基土质较差，需整体换填，且换填后还有三趟雨污水管道施工，施工难度较大，周期较	2019/5/9 9:49:10	1
提高A市民营幼儿园老师抨障民办幼儿园教职工待遇，民办幼儿园聘任教职工要依法签订劳动合同，依法缴纳城镇企业职工养老保险	2019/5/9 9:49:14	1
公寓能享受人才新政购房?含机关事业单位在编人员），年龄35周岁以下（含），首次购房后，可分别申请6万元、3万元的购房补贴	2019/5/9 9:49:42	1
'A市公交站点名称变更的;”，原“马坡岭小学”取消，保留“马坡岭”的问题。公交站点的设置需要方便周边的市民出行，现有公	2019/5/9 9:51:30	1
A3区含浦镇马路卫生很差·别是学士街道和含浦街道，鉴于您问题中没有说明卫生较差的具体路段，也没有相应的参照物，同时您也	2019/5/9 10:02:08	1
教师村小区盼望早日安装彳的宜居水平，2018年6月7日，A市A3区人民政府办公室下发了《关于A市A3区既有多层住宅增设电梯实施	2019/5/9 10:18:58	1
K东澜湾社区居民的集体闷要求。区教育局已启动二次装修前期准备及设施设备采购等工作。下一步，街道将督促开发商尽快完成	2019/1/29 10:53:00	1
麓阳光住宅楼无故停工以,工。2018年11月20日，在责任单位落实分户检查后，西地省楚江新区建设工程质量安全督查站监督人员召	2019/1/16 15:29:43	1
和顺路洋湖壹号小区路段彳车道两侧裁种了行道树，其余部分也按规划要求完成了建设，其中西边绿化带面积约6000平方米，由于	2019/1/16 15:31:05	1
2区大托街道大托新村违玄土地权属归该公司），由该公司支付一笔耕地征收补偿款给原大托村，但截至目前，该公司未能支付任何	2019/3/11 16:06:33	1
鄱阳村D区安置房人防工程(、二期c区已建，需补办人防手续，按长人防发[2014]7号文件要求，鄱阳村三期https://baidu.com/平方	2019/1/29 10:52:01	1
K段请求修建一座人行天桐牵头，区城乡建设局、区规划分局配合进行具体选址，招标（邀标）进行方案设计等，尽快启动万国城	2019/1/14 14:34:58	1
报A市芒果金融平台涉嫌诈情况回复如下：经查，您所反映的相关警情，已由银盆岭派出所立刑事案件侦查，案件正在侦办中。感谢	2019/1/3 14:03:07	1

图 3.2.2 标注表

3.3 相关度

3.3.1 词袋模型语料库

经过预处理后从附件得出的文本就变成了一个个特征词，建造一个词袋模型就是将所有的特征词放进 y 一个袋子里，然后统计出现的频率，计算。这样的特征词是没有上下文以及先后顺序的。

词袋模型通常分三步，分词，统计特征值，标准化。

通常分词后会建成一个字典，即词典索引。这个字典的 key 是特征词，value 为索引号。

之后会进行一个统计词频步骤，统计后本题将会得到该文本所有词的词频，此时，这个文本就转换成了一个词向量，向量的维度由不同特征词获得。

本题要获取文本的词袋向量构成语料库，例如：

```
[[(0, 2), (1, 1), (2, 1), (3, 1), (4, 1), (5, 4), (6, 1), (7, 2), (8, 1), (9, 1), (10, 1), (11, 1), (12, 1), (13, 1), (14, 1), (15, 1), (1
6, 1), (17, 1), (18, 1), (19, 1), (20, 1), (21, 1), (22, 1), (23, 1), (24, 2), (25, 2), (26, 1), (27, 1), (28, 1), (29, 1), (30, 1), (31,
```

图 3.3.1 词袋模型部分结果

3.3.2 TF-IDF 模型

使用 TF-IDF 将文本转换为加权词频矩阵，原理见上文。

3.3.3 计算相似度

特征向量化后通过距离公式计算文本相似度。

（一） 欧氏距离

欧几里得距离，是常见的距离度量，计算的是两个点在多为空间中的绝对距离。得出结果越小，相似度越大。[9]

其公式为：

$$\mathrm{dist(X,Y)} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

（二） 余弦相似度

余弦相似度是计算两个向量点的夹角的余弦值，以此作为两个向量点的差异。跟欧氏距离相比注重的更加是方向上的差异。余弦值越接近 1，相似度越大。

例如 A,B 两个向量点。

计算公式为：

$$\cos\theta = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sum_{i=1}^{n}(A_I)^2 \times \sum_{i=1}^{n}(B_i)^2} = \frac{A^T \times B}{\|A\| \times \|B\|}$$

因为附件 4 中的[留言详情]跟[答复意见]两个文本的长度各不相同，所以本题计算相似度时采用了余弦相似度计算，只考虑两个文本之间方向上的差异，不考虑长度。

3.3.4 相关性结果展示

部分结果展示如下：

中	网友：您好!您所反映的问题，已转交。2019年10月17日	2019/10/1	差
个	网友：您好!您所反映的问题，已转交。2019年9月29日“UU008548”您好！您在201	2019/9/2	良
乡	网友：您好，您所反映的问题，已转交许家坊地家族乡作出回应。2019年6月24日“	2019/6/2	良
3	“UU0081008”您好！您在2019年3月11日《问政西地省》上发帖反映的问题，我局高	2019/3/1	良
一	“UU0082351”您好，您所反映的问题已转交县交通运输管理局处置。2018年4月17日	2018/4/1	良
下，	网友你好：您反映的问题，镇相关负责人经过认真调查，现将情况回复如下：当前‘	2019/11/1	良
导，	您好，你所反映的问题已转交相关单位调查处置。	2019/10/	差
这	您好，你所反映的问题已转交相关单位调查处置。	2019/10/	差
水	您好，你所反映的问题已转交相关单位调查处置。	2019/10/	差

图 3.3.4 相关性结果展示

3.4 可解释性

3.4.1 基于词向量的朴素贝叶斯的分类器

3.4.1.1 朴素贝叶斯分类器原理

贝叶斯决策论是概率框架下的一个实施决策的基础方法。在所有相关概率知道的理想情况下，贝叶斯决策论会考虑基于这些概率和误判来选择最优的类别结果。

但是在现实中的训练里面，因为自变量各个维度上的组合方式是呈指数式增长的，会远远的大于样本数量，就会导致很多可能的样本取值从未在样本集中出现。

所以为了避免贝叶斯公式的训练障碍，朴素贝叶斯采用了“属性条件独立假设(attribute conditional independence assumption)”，对一直类别，假设所有的属性相互独立，每个属性各自独立地对分类结果产生影响。

其表达式为：

$$h_{nb}(x) = arg\max_{c\in y} P(c)\prod_{i=1}^{d} P(x_i|c)$$

其中 d 表示属性的个数，x_i表示 x 在 i 个属性的取值，因为 p(x)由样本集唯一确定，即对所有类别的 p(x)都相同。

朴素贝叶斯的分类器训练过程是基于训练集 D 来估计类先验概率 P(c)，并为每个属性估计条件概率 $p(x_i|c)$，用 D_c 表示训练集 D 中 c 类样本组成的集合，如果有充足的独立同分布样本，则可以很容易估计先验概率[10]：$p(c) = \frac{|D_c|}{|D|}$

对于离散属性，D_{c,x_i} 表示 D_c 在 i 个属性上取值为 x_i 的样本集合，条件概率 $P(x_i|c)$为：

$$p(X_i|c) = \frac{|D_{c,x_i}|}{|D_c|}$$

对于连续属性，假如 $P(x_i|c) \sim N(\mu_{c,i},\sigma^2_{c,i})$，其中$\mu_{c,i}$，$\sigma^2_{c,i}$分别表示为第 c 类样本在属性上的均值与方差，则 $P(x_i|c)$为：

$$P(x_i|c) = \frac{1}{\sqrt{2\pi}\sigma_{c,i}} \exp\left(-\frac{(x_i - u_{c,i})^2}{2\sigma^2_{c,i}}\right)$$

3.4.1.2 朴素贝叶斯分类

放入数据进朴素贝叶斯分类模型，进行训练获得如下部分分类结果：

```
array(['1', '1', '1', '1', '1', '1', '1', '1', '0', '1', '1', '1', '1',
       '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1',
       '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1',
       '0', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1',
       '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '0', '1', '1',
       '0', '1', '1', '1', '1', '1', '1', '0', '1', '1', '1', '1', '1',
       '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1', '1',
       '1', '1', '1', '1', '1', '1', '0', '1', '1', '1', '1', '1', '1',
```

图 3.4.1 朴素贝叶斯分类器部分训练结果

3.4.2 分类模型评估

将数据按 8：2 拆分成训练集和验证集，在模型训练后，需要对其结果进行评价，此时需要引入评价指标。

3.4.2.1 混淆矩阵验证

混淆矩阵是用于分类模型评估中的一个最基本、最直接的方法。它会分别统计分类模型归错类，归对类的观测值。

混淆矩阵的一级指标 y 有四个基础指标，TP,FN,FP,TN。

- TP(True Positive),真实值是 Positive，模型预测为 Positive。

• FN(Flase Negative),真实值是 Positive，模型预测为 Negative。

• FP(Flase Positive)，真实值是 Negative，模型预测为 Positive。

• TN(True Negative)，真实值是 Negative，模型预测为 Negative。

将其四个一级指标放在一张表格中获得的矩阵被称为混淆矩阵。

混淆矩阵		真实值	
		Positive	Negative
预测值	Positive	TP	FP
	Negative	FN	TN

3.4.2.2 混淆矩阵的指标与评估

混淆矩阵得出的结果是 TP、TN 越大，FP、FN 越小，此时评估的结果越理想。

将训练数据带入模型后，将结果放进 python 第三方库 sklearn 的 confusion_matrix 函数中，获得部分结果如下：

```
from sklearn import metrics
metrics.confusion_matrix(Y_test, Y_pred)
```

```
array([[ 41,  19],
       [ 11, 493]], dtype=int64)
```

图 3.4.2 混淆矩阵结果

3.4.2.3 F1-Score 评估

调用 python 的第三方库 sklearn 中的 f1_score 函数，设置参数为“micro-F1”，评价分数如下：

```
# 查看分数
from sklearn.metrics import f1_score
print ("模型的f1_score: %0.3f " % f1_score(Y_test,Y_pred , average='micro'))
```

```
模型的f1_score: 0.947
```

图 3.4.2 F1-Score 评估结果

3.4.3 可解释性结果展示

部分结果展示如下：

网友：您好!您所反映的问题，已转交。2019年10月17日	2019/10/1	差	0
网友：您好!您所反映的问题，已转交。2019年9月29日“UU008548”您好！您在2019年9月27日《	2019/9/2	良	1
网友：您好，您所反映的问题，已转交许家坊地家族乡作出回应。2019年6月24日“UU008485”	2019/6/2	良	1
“UU0081008”您好！您在2019年3月11日《问政西地省》上发帖反映的问题，我局高度重视，迅速	2019/3/1	良	1
“UU0082351”您好，您所反映的问题已转交县交通运输管理局处置。2018年4月17日尊敬的“时刻	2018/4/1	良	1
网友你好：您反映的问题，镇相关负责人经过认真调查，现将情况回复如下：当前“村村响”主要	2019/11/1	良	1
您好，你所反映的问题已转交相关单位调查处置。	2019/10/2	差	0
您好，你所反映的问题已转交相关单位调查处置。	2019/10/2	差	0
您好，你所反映的问题已转交相关单位调查处置。	2019/10/2	差	0

图 3.4.3 可解释性部分结果(0，1 是结果)

4 参考资料：

[1]石凤贵.基于 TF-IDF 中文文本分类实现[J].现代计算机,2020(06):51-54+75.
[2]梁华,宋玉龙,钱锋,宋策.基于深度学习的航空对地小目标检测[J].液晶与显示,2018,33(09):793-800.
[3]张洪胜,高海宾.基于模拟样本训练的支持向量机[J].韶关学院学报,2019,40(12):13-17.
[4]周志华著.机器学习[M].北京：清华大学出版社.2016.
[5]杨锋.基于线性支持向量机的文本分类应用研究[J].信息技术与信息化,2020(03):146-148.
[6]黄宗财,仇培元,陆锋,吴升.基于联合主题特征的网络新闻文本蕴含环境污染事件检测[J].地球信息科学学报,2019,21(10):1510-1517.
[7]丁世飞，靳奉祥.现代数据分析与信息模式识别[M].北京：科学出版社.2019.
[8] 石胜飞.大数据分析与挖掘[M].北京：人民邮电出版社.2018.
[9]王博生,何先波,朱广林,郭军平,陶卫国,李丽.基于近邻协同过滤算法的相似度计算方法研究[J].绵阳师范学院学报,2019,38(08):84-90.
[10]王鹤琴,王杨.基于贝叶斯决策的网格社区案卷分发模型[J].山东大学学报(理学版),2018,53(11):85-94.