基于自然语言处理的智慧政务系统分析

摘要

近些年来，深度学习与自然语言处理技术发展迅速，并且被广泛的应用。本文基于“智慧政务”中的文本挖掘的应用，构建了逻辑回归模型（LR 模型）、TextCNN,TextRNN 等分类模型对市民反应的留言问题进行对比分类;构建 DBSCAN 模型对热点问题进行聚类，并定义热度指标给予排名。

对于第一问，首先，对数据进行预处理。由于题目中所给的数据源文件为中文留言文本，所以首先对其进行 Jieba 分词和去停用词。将分词好的中文词语通过 One-Hot 进行编码，通过 Word2Vec 模型中的 CBOW 模型进行特征提取，获得初步词向量表示。题目中要求对留言内容基于一级标签进行分类，所以我们选取了三个分类模型，分别是逻辑回归模型、TextCNN 模型，TextRNN 模型。选取这三个模型分别进行分类实验,通过四项指标召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$的值对模型进行评价，并得出其对应模型的混淆矩阵，比较每个模型所对应的那组评价指标的数值，选取最优分类模型。通过测试三类模型，可看出 TextCNN 较逻辑回归模型的ACC提升了 17.3%,$F1-score$提升了 17.2%,较 TextRNN 的ACC提升了 0.4%，$F1-score$提升了 0.1%，性能显著提高，所以基于问题一选取的是 TextCNN 对留言内容进行一级标签分类。

对于第二问，是热点问题的挖掘，先进行分词处理，之后通过基于词频统计的 TF-IDF 模型对关键词抽取，设定阈值m，求出$TFIDF(w_i)$的值，当$TFIDF(w_i)>m$，从而将这些实词w_i作为留言文本中的特征词，构成了留言文本的特征向量，降低了模型的维度。之后通过余弦相似度的计算，得出每个特征词和其他词汇之间的相似度，从而反应文本相似度。之后通过 DBSCAN 聚类算法，以扫描半径（eps)为0.25，最小包含点数($minpoints$)为 5 进行迭代一百次，对留言完成聚类，利用基于互信息的方法来衡量聚类效果，MI与NMI取值范围是[0,1], AMI取值范围是[−1,1],他们都是值越大意味着聚类越符合。本题中通过计算得出$AMI=0.805$结果较高，说明聚类效果较好。为了更好的定义留言热点问题的热度评价指标，我们首先将留言文本定义为一个 7 元组W：

$W=（number,user,subject,text,time,attitude\_num,against\_num）$，我们将热度评价指标定义为$topic_hot$ 指标，通过留言出现的次数、平均点赞数、平均反对数三个因子进行加权求和，得出$topic_hot$ 指标的值，比较其大小，求出前五名的热点问题。

对于第三问，此题为开放性问题，我们首先提取影响留言答复意见的评价指标，然后构建答复意见质量的评价指标体系与模型，最后验证模型性能。本题中我们抽取了七个评价指标，分别为：完整性（$Complete$）、一致性（$Coherent$）、可信性（$Credibility$）、可解释型（$Readability$）、相关度（$Relevancy$）、可

解释型（$Readability$）、时效性（$Efficiency$）构建答复意见质量的评价指标体系，用广义线性模型（GLM）回归方法构建答复意见评价模型。用$Quality$表示答复意见质量，计算Q的值，并设定一个等级标准来衡量政府的答复意见质量。

上述三个问题都基于 Python 编程实现。

关键词：Jieba Word2Vec TextCNN TextRNN TF-IDF DBSCAN 广义线性模型 One-Hot

Abstract

In recent years, deep learning and natural language processing technologies have developed rapidly and been widely used. In this paper, based on the application of text mining in "smart government", a Logistic Regression model (LR), TextCNN, TextRNN and other classification models are constructed to compare and classify the message problems of citizens' reactions. DBSCAN model was constructed to cluster hot issues, and heat index was defined to rank them.

For the first question, first, we preprocess the data. Since the data source file given in the title is the Chinese message text, it is first divided into Jieba words and stop words. Then the Chinese words with good word segmentation are encoded by one-hot, carried out feature extraction by CBOW model in Word2Vec model, the initial word vector representation is obtained. In the topic, it is required to classify the message contents based on first-class labels, so we choose three classification models, namely Logistic Regression model, TextCNN model and TextRNN model. The three models were selected for classification experiments. The models were evaluated by the values of $Recall$, $Precision$, ACC and $F1-score$, and the confusion matrix of the corresponding models was obtained. The values of the corresponding group of evaluation indexes of each model were compared, and the optimal classification model was selected. Through testing the three models, it can be seen that TextCNN improves by 17.3% and $F1-score$ by 17.2% compared with ACC of Logistic Regression model, 0.4% and 0.1% compared with ACC of TextRNN, and its performance is significantly improved. Therefore, TextCNN is selected based on question 1 to carry out primary label classification for message content.

Asked for a second, is the hot issue of digging, the first word segmentation processing, after the $TF-IDF$ model based on word frequency statistics for keyword extraction, set a threshold value m, get the value of $TFIDF(w_i)$, when $TFIDF(w_i) > m$, thus the content words w_i as key words in the text, constitutes the message text eigenvector, reduce the dimension of the model. Then, through the calculation of cosine similarity, the similarity between each feature word and other words is obtained, so as to reflect the similarity of text. After that, the DBSCAN clustering algorithm was used to iterate for 100 times with the scan radius (eps) of 0.25 and the minimum inclusion point ($minpoints$) of 5. The clustering of messages was completed, and the method based on mutual information was used to measure the clustering effect. The value range of MI and NMI was [0,1], and the value range of AMI was $[-1,1]$.In this problem, it is calculated that AMI is equal to 0.805 and the result is higher, indicating that the clustering effect is better. In order to better define the heat evaluation index of the message hot issue, we first define the message text as a 7-tuple W:

$W = (number, user, subject, text, time, attitude\_num, against\_num)$, We define the heat evaluation index as the $topic\_hot$ index, and calculate the value of the $topic\_hot$ index by the weighted sum of the three factors of the number of

comments, the average thumb up number and the average negative number, and then calculate the top five hot issues.

For the third question, which is called an open question, we first extract the evaluation indexes that affect the comments, then build the evaluation index system and model for the quality of comments, and finally verify the performance of the model. Subject of evaluation index, we extract the seven are: Complete, Coherent, Credibility, can be Readability, Relevancy, can be Readability and Timeliness build a reply opinion quality evaluation index system, evaluation model using Generalized linear model（GLM） regression method to build a reply opinion. Use "$Quality$" to represent the Quality of the replies, calculate the value of "Q", and set a rating standard to measure the Quality of the government's replies.

All three questions are based on Python.

Keywords: Jieba Word2Vec TextCNN TextRNN TF-IDF DBSCAN Generalized linear model one-hot

目录

一、简介

1.1 挖掘意义

社情民意调查是采用科学的调查和统计学方法，对一定时期，一定范围内的社会公众对所存在的社会现实的主观反应的调查，具有反映民意，引导舆论，为政府决策提供参考，检验政策实效等作用。我国的社会民情调查起步较晚，在改革开放后才逐步发展起来。随着我国近几年经济的高速发展，社情民意调查成为了政府部门和人民群众沟通的桥梁。近年来，随着互联网的发展，对社情民意信息获取的途径不断增加，微博，微信，市长信箱，阳光热线也成为了网络问政平台，成为了政府了解民意、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增加，这就给相关部门的留言划分以及热点整理工作带来了挑战。比如，对于海量留言数据，单纯依靠人工划分，耗力耗时，效率低下，而且分类结果主观性强，不同的人进行分类，结果可能不相同；再就是部分留言可能出现相似或相近的关键词，那么就需要对其重新分类。等等这些问题，面对海量文本数据无疑是对人类的挑战。因此，我们希望通过建立基于自然语言处理技术的“智慧政务”系统，帮助政府部门提高管理水平和处理问题的效率。

为了构建智能文本分类模型，对留言更好的分类，学术界对文本挖掘的研究从未止步。自然语言处理目标是使机器在能够理解文本的情况下，正确的对文本进行分类，以节省人力物力财力，对“智慧政务”系统的构建具有重要的研究与应用价值。

1.2 挖掘目标

我们要构建一个智能的文本分类模型。模型可以持续将群众留言分派给相应职能部门处理。那么针对于问题一的情景中，对于收集到的群众的留言收集到的问题，模型可以根据提取的留言中的关键词依据所提供的分类标签对留言进行一级标签分类。

更行形式化地，我们将要分类的留言分解为一个个词向量对模型进行训练，我们需要再留言中提取关键词，为了解决这个问题需要将所有文本分词映射到数值空间中，将文字转化成数字进行处理。

1.3 挖掘流程

本题的挖掘主要分为两大部分，预处理部分和对建立的分类模型评价部分。

其中预处理部分包括分词，去停用词，通过 Word2Vec 将中文词组转化为词向量。建立关于留言内容的一级标签分类模型并给予模型评价是关键步骤，常用的文本分类算法有逻辑回归算法、朴素贝叶斯和神经网络算法。本题采用的是逻辑回归模型，通过设置概率阈值实现其分类功能，从而对留言内容进行分类。

二、预处理

2.1 分词

无论是在汉语还是英语中，词一般都代表着最小的语义单位，题目中所给数据是中文文本，中文文本的特点是词与词之间没有明显的界限，因此在研究中就需要将句子划分成词，才可以继续转入后续的研究分析中。在 Python 语言中，中文分词一般选用 Jieba 分词器，对收集到的每一条留言进行中文分词。

Jieba 分词是基于统计的分词方法，采用精确的分词模式，试图将句子最精确地切开，以用于文本分析。对训练集的。。。条样本数据循环遍历，使用 Jieba 库的 cut 方法获得分词列表赋值给变量。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），每个词条对应图中一条有向边，词频可以看作 DAG 中边的权重，基于词频计算出当前字到句尾的概率，有了最大概率路径，分词结果也就确定。

2.2 去停用词

在文本处理中，经常遇到一些几乎在所有文本中都会出现，不具备特殊性和区分度，这样的词反而会稀释那些有区分度的词。它们大多是一些单字，单字母以及高频的单词，我们称这种词为停用词。比如大多数中文文本中的“的、得、地、你、我、他”等，英文中的“the、this、that、an、a”等。对于停用词一般在对文本数据清洗的时候就将其过滤删除，这样可以减小特征词的数量，从而可以提高文本分类的准确性。

2.3 Word2Vec

为了更好的处理文本信息，从而分析建立模型，我们首先将所给的中文留言文本转化为计算机能够理解的数字形式。我们将要分类的留言分解为一个个词向量对模型进行训练，需要将所有文本分词映射到数值空间中，词向量的维度就是

代表词的大小，同时采用了 One-Hot（独热编码）的形式，使词向量大部分都是 0，只有一个维度是 1，那么这个维度就代表了当前的词。

Word2Vec 实质上是一个浅层神经网络模型。可以在海量数据集和高维词典中进行有效的计算，通过对输入的文本集计算上下文与目标词之间的概率关系而得到词语的词向量，该词向量很好的保留了上下文的语义，能计算词语之间的相似性，作为分类模型的特征项，已经广泛应用于自然语言处理的多个方面。Word2Vec 通过 CBOW 和 Skip-Gram 两种模型定义数据的输入和输出。CBOW 模型是利用上下文预测当前词，而 Skip-Gram 是利用当前词预测上下文。

三、问题分析

3.1 建立关于留言内容的一级标签分类模型

根据前面已经预处理好的数据，我们分别采用逻辑回归模型、TextCNN 和 TextRNN 模型分别基于留言内容对留言进行分类，同时进行比较两种模型的精确度，选取更优的模型。

由附件 2 给出的数据，可以看出一级标签一共有 7 类，分别是：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。根据每个标签特征进行分类。

3.1.1 预处理

（1）Jieba 分词和去停用词：由于附件 2 中给的是中文的留言文本，存在句子冗杂，主题不明确，省略，重复等问题，有较强的干扰性和隐蔽性，对分类识别有巨大的挑战。因此首先对其进行预处理工作。根据预先设计好的预处理程序对其完成去干扰符号和空格（包括全角情况下的空格）、去停用词以及对电话、价格、日期等数字，英文字母的去除。再就是反应的问题的重复性，一个问题多次出现会对文本处理造成干扰，以及部分文本的无意义表达太多，或者是文本语义带来的词语交叉都是预处理环节所要解决的问题。

例如：分词之前：“彻底解决用水难的问题”进行 Jieba 分词，分词结果：“彻底”“解决”“用水”“难”“问题”。

由此可见当进行预处理之后，所得到的即为一些分词短语构成一个集合 T，降低了噪声，为下一步实现 Word2Vec 提高了生成词向量的准确性。

（2）词云图是文本结果展示的有利工具，通过词云图，可以展示对文本进行分词后出现的高频词，从而让人一目了然，起到强调的结果，使得读者一眼就可以得到主旨信息。如图 1，为教育文体类词云图，可以直观的看出文本的分词结果

中的关键词，如：“学校”、“学生”、“教育局”、“招生”等。

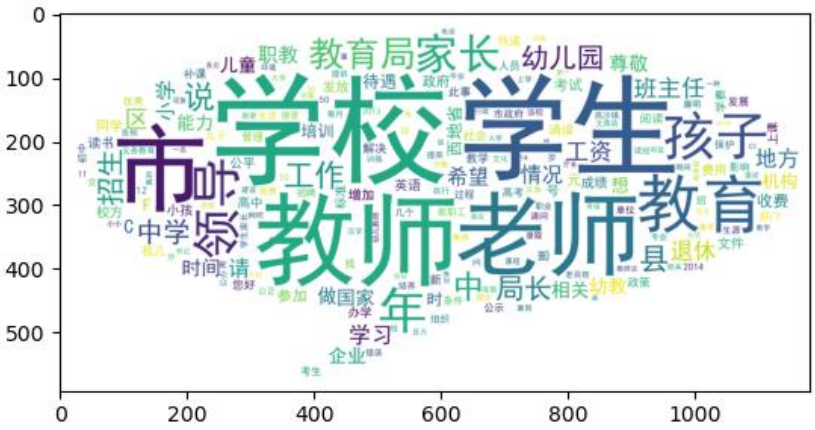

图 1　教育文体类词云图

3.1.2 基于 Word2Vec 留言特征提取

本题采用了自然语言处理的 Word2Vec 方法，通过 Word2Vec 工具分别对预处理后的留言文本进行词向量训练。我们将所有文本分词映射到数值空间中，词向量的维度就是代表词的大小，同时采用了 One-Hot（独热编码）的形式，实现从非结构化数据到结构化数据的转化，将每个分词表示成一个个长长的词向量，使词向量大部分都是 0，只有一个维度是 1，那么这个维度就代表了当前的词。

1）Word2Vec 的 CBOW 模型

之后通过 Word2Vec 中的 CBOW 模型，其工作原理是通过上下文预测当前中心词，如图 2 所示：

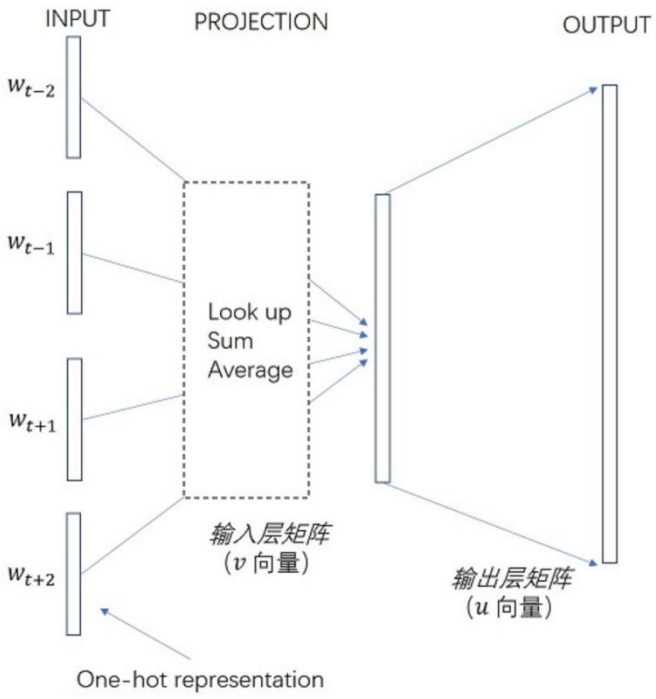

图 2　Word2Vec CBOW 模型

通过图 2 的结构示意图中，$w(t-1),w(t-2),w(t+1),w(t+2)$表示输入的词向量，实际上是一个 One-Hot 的编码，通过构建一个词典，实现分词的索引，在这里词向量大部分都是 0,只有一个维度是 1,那么这个维度就代表了当前的词。在本题中，我们设置了$widows_size$为 3，向量的维度为 200 维，现在将中心词w_i的上下文分词的独热编码输入，由于 widows_size=3，所以有 2 个 One-Hot 编码向量输入。设输入层的权值矩阵为 W(i 行 n 列)，v 为 W 的一行，则有 lookup 的过程：

$$w_i^T W_{in} = v_i \qquad (1)$$

由此可知，将 One-Hot 编码的向量w_i和 W 相乘，由于w_i中只有一个维度是 1，所以得出了权值矩阵中w_i对应的每一行向量，则该行向量v_i就为分词w_i对应的词向量。

2）特征提取处理

通过上文中的 lookup，我们得出了每一个分词的词向量，由于这些词向量是权值矩阵W的对应的每一行分向量，所以维数相同，将其累加后求取平均值所得向量就是该句对应的语义向量作为最终的数据输入分类模型。

我们使用的 CBOW 模型的训练目标就是使训练集中的样本的似然概率最大，似然概率如下：

$$\prod_{i=1}^{T} P(w_i|w_{i-2},w_{i-1},w_{i+1},w_{i+2}) \qquad (2)$$

该式为 CBOW 的目标函数，为了找到最优解，需要优化似然函数的值，则有：

$$-log\sum_{i=1}^{T}P(w_i|w_{i-2},w_{i-1},w_{i+1},w_{i+2}) \qquad (3)$$

即使用梯度下降法，多次迭代求取最优值。

3.1.3 构建分类模型

常用的文本分类算法有逻辑回归算法和神经网络算法。本题中我们选取逻辑回归算法（LR）和神经网络算法中的 TextCNN 以及 TextRNN 对其分类功能进行对比，选取最优模型对留言内容进行分类。下图是文本训练流程图：

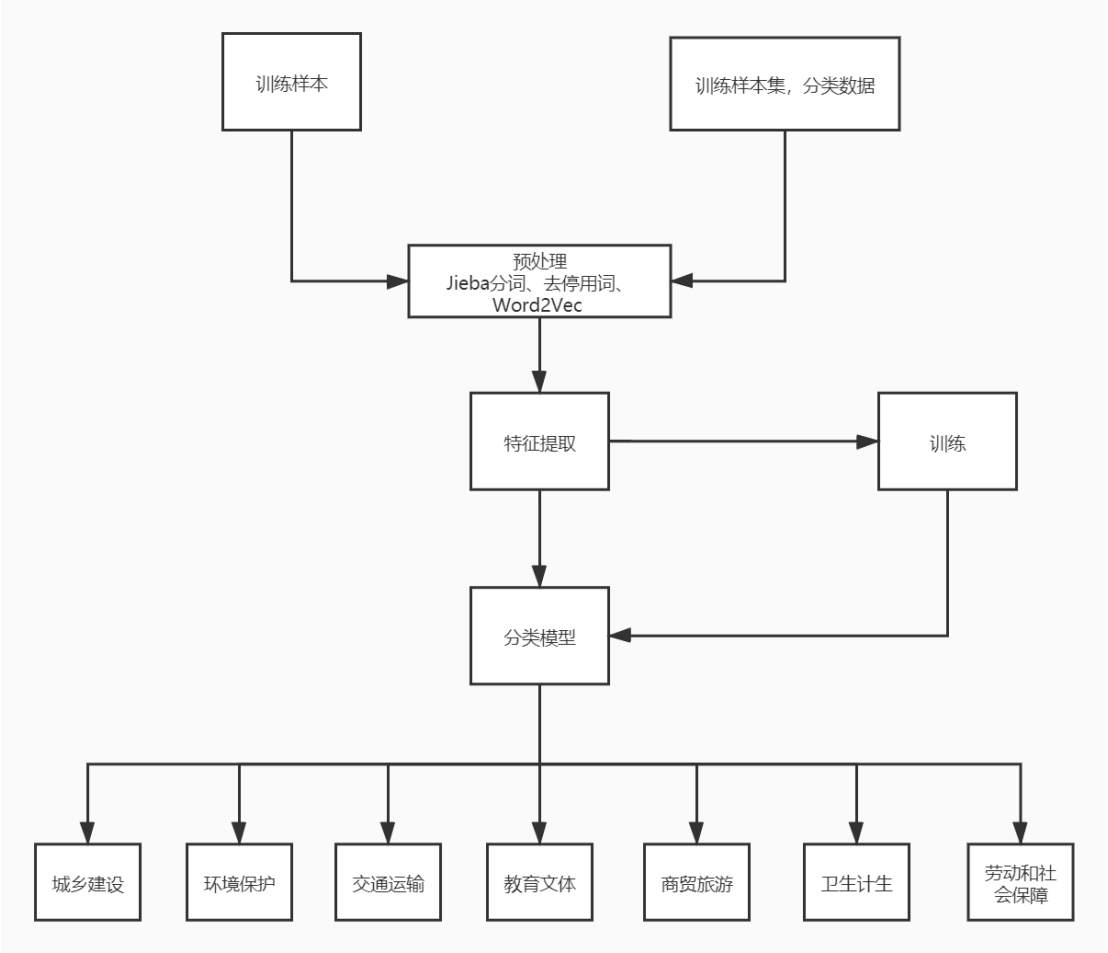

图 3 文本训练流程图

本数据集有七个类别，是一个七分类问题，召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$的值对应的是二分类问题中的评价标准，因此，将七分类问题转换为多个二分类问题。在每一轮实验中，将七种类别两两作为一类，共有 6 种组合，对所有组合进行模型训练测试。训练过程中，部分类别的存在数据不均衡现象，通过样本均衡技术减小数据不均衡的问题。四个评价指标中，$F1-score$ 为首要指标。在每一轮实

验中，选取最好的分类结果作为这一轮实验的最终结果。

3.1.3.1.LR 模型

逻辑回归算法是典型的二分类算法，既可以用于预测又可以用于分类。LR 模型可以被看作一个 Sigmoid 函数，研究所选取的 Sigmoid 函数，将其输入的向量映射为概率值，实现预测功能，再通过调整其阈值，从而进行分类。

（1）LR 模型的理论依据：是广义的线性回归模型，由线性回归的基本思想中的判别公式：

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n = \theta^T x \quad (4)$$

以线性回归的判别公式为基础得出逻辑回归公式：

$$h_\theta(x) = \frac{1}{1 + e^{-z}} = \frac{1}{1 + e^{-\theta^T x}} \quad (5)$$

上式就是$Sigmoid$函数，通过$Sigmoid$函数引入了非线性映射，从而可以使线性回归映射到$0\sim1$之间，值域的中间值为 0.5，由此可以知道$h_\theta(x)$的值域也是（0，1），有利于直观的判断。所以对于二分类问题，需要将测试样本的n个特征值具体数据x代入$h_\theta(x)$中计算，按照 Sigmoid 函数形式求出

$$P(y = 1|x;\theta) = \frac{1}{1+e^{-\theta^T x}}, \quad (6)$$

从而进行分类。一般我们规定：$h_\theta(x) < 0.5$时，则说明当前处理的数据属于同一类别；相反的，就属于另一类。随着不断的实验验证，这里的阈值我们设为了 0.5 。

（2）实验结果

通过将 Word2Vec 处理的词向量最后生成的向量输入 LR 模型中，可以得到其四项指标召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$所对应的值。同时通过混淆矩阵进行表示：

表 1 分类结果混淆矩阵

真实情况	预测结果	
	正例	反例
正例	TP	FN
反例	FP	TN

召回率反映了正例中被正确分类的概率，其表达式为

$$R = \frac{TP}{TP+FN} \quad (7);$$

精确度反映了分类为正例中被正确分类的概率，其表达式为

$$P = \frac{TP}{TP+FP} \quad (8);$$

准确率是分类任务中最常用的、最基本、最重要的一个评价指标，其表达式为

$$ACC = \frac{TP + TN}{TP + FN + Fp + TN} \quad (9)$$

F1-score 是基于召回率和精确率的调和平均，其表达式为：

$$F_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2P_{\dot{z}}p_i}{P_i + R_i} = \frac{2TP}{2TP - TN + FP + FN} \quad (10)$$

基于上述四个指标，通过 LR 模型可得：

表 2　LR 四项指标

模型	召回率	精确度	准确率	$F1-score$
LR 模型	0.737	0.741	0.735	0.738

同时可得出其混淆矩阵如图所示：

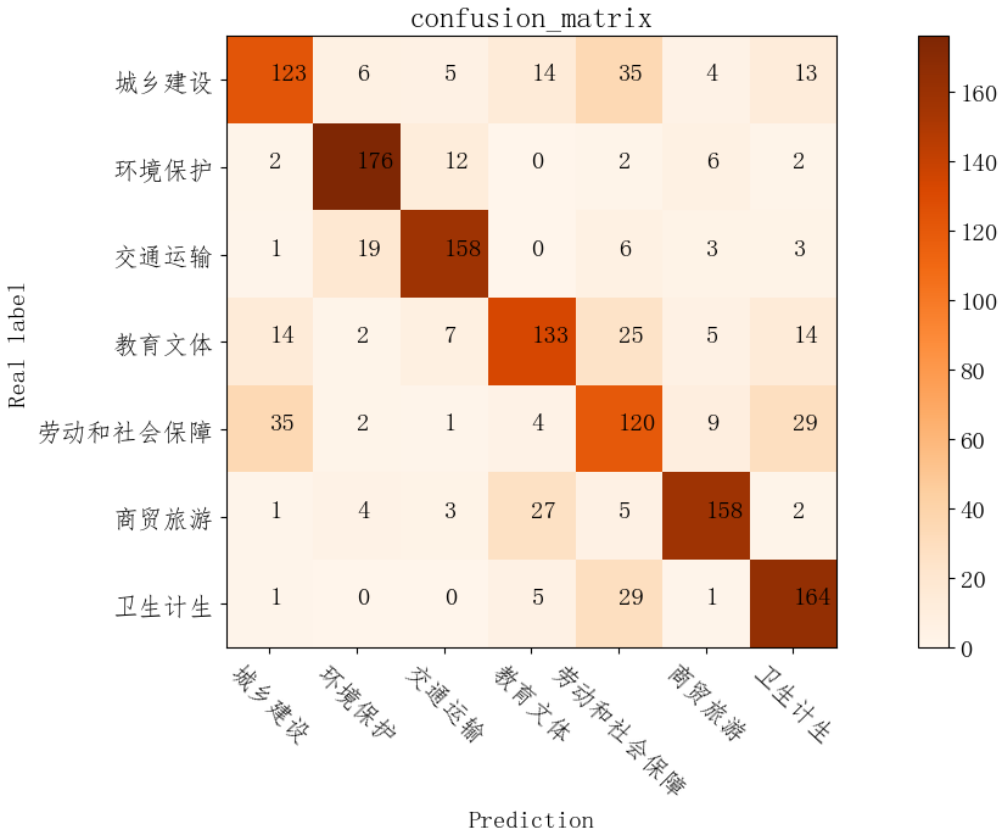

图 4 基于 LR 模型混淆矩阵

由上述实验结果来看，逻辑回归模型的准确率仅有 73.5%，相对较低。

3.1.3.2 TextCNN 模型

除了 LR 模型之外，本题还采用了卷积神经网络模型即 TextCNN 进行对留言内容分类，通过进行 10 次随机实验求取平均值，最后得出四个评价指标的值。

TextCNN 模型是 2014 年 Yoon Kim 针对 CNN 的输入层做了一些变形，提出了此模型，TextCNN 在网络结构上没有任何变化，只有一层卷积，一层 max-pooling，最后将输出 softmax 进行 7 分类。工作原理图如图所示：

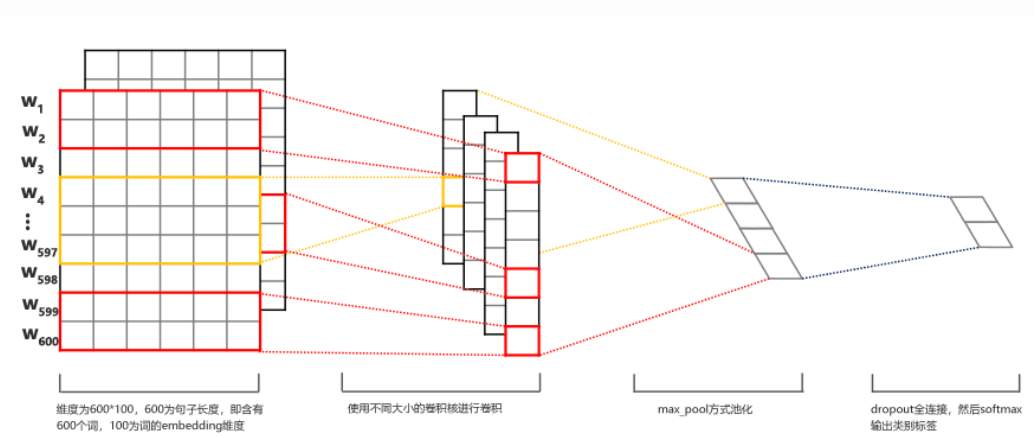

图 5　TextCNN 原理图

本题中给出的留言文本为一维数据，通过 Word2Vec 这种 embadding 方式，将每个分词映射成一个词向量，将自然语言数值化，方便后续处理，将所有的词向量处理完成之后构建一个二维矩阵，最为最初的输入。

由原理图可知，整个模型由四部分组成：输入层、卷积层、池化层、全连接层。卷积操作是 CNN 的重要特征之一，卷积层以特征映射为组织的方式，其中的每一个单位与前一层的局部感受野相连接，利用共享的卷积核（或称过滤器）与局部感受野做卷积运算，这里的卷积核为 128，再通过激活函数做非线性运算，得到特征值，给定一个矩阵$X \in \mathbb{R}^{M\times N}$，和卷积核$F \in \mathbb{R}^{M\times N}$，一般$m << M$，$n \ll N$，其卷积如式所示：

$$ConV_{ij} = \sum_{u=1}^{m}\sum_{v=1}^{n} f_{uv} \cdot x_{i\cdot u+1:j-v+1} \qquad (11)$$

通过卷积操作，将矩阵映射为一个$3*1$的特征矩阵$[2,3,4]$,通过$max-pod$方式池化，从提取的特征矩阵中选取最大值，用$dropout$跟其他通道的最大值拼接，组合成筛选过的的特征向量，之后再通过 softmax 层对文本进行 7 分类。

通过进行 10 次 TextCNN 进行对留言内容分类验证，分别得到 10 组召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$ 的值，对其求取平均值则有：

表 3　TextCNN 四项指标

模型	召回率	精确度	准确率	$F1-score$
TextCNN	0.912	0.923	0.908	0.910

并得出其混淆矩阵：

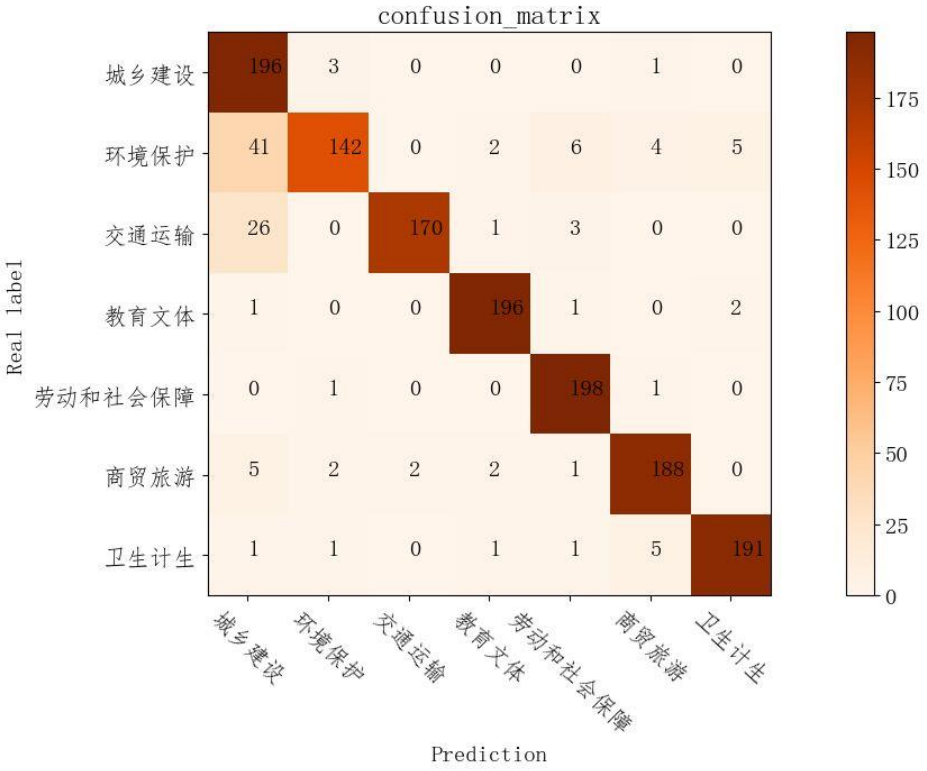

图 6　TextCNN 混淆矩阵

3.1.3.3 TextRNN 模型

TextRNN 模型也是进行文本分类的常用的一种模型。TextRNN 指的是利用 RNN 循环神经网络解决文本分类问题。所以在本题中，我们除了采用 LR 模型、TextCNN 模型之外，还采用了 TextRNN 对留言文本进行分类。

1.TextRNN 原理

TextRNN 是直接利用循环神经网络处理文本的词向量序列，并通过 softmax 函数进行分类. 这里的文本可以一个句子，文档或篇章，因此每段文本的长度都不尽相同。在对文本进行分类时，我们一般会指定一个固定的输入序列或文本长度：该长度可以是最长文本或序列的长度，此时其他所有文本都要进行填充以达到该长度；该长度也可以是训练集中所有文本长度的均值，此时对于过长的文本需要进行截断，过短的文本则进行填充。

2.两种典型结构

（1）Structure 1

流程：

$$embedding——> BiLSTM——> concat\ final\ output/average\ all\ output———> softmax\ layer$$

结构图如下图所示：

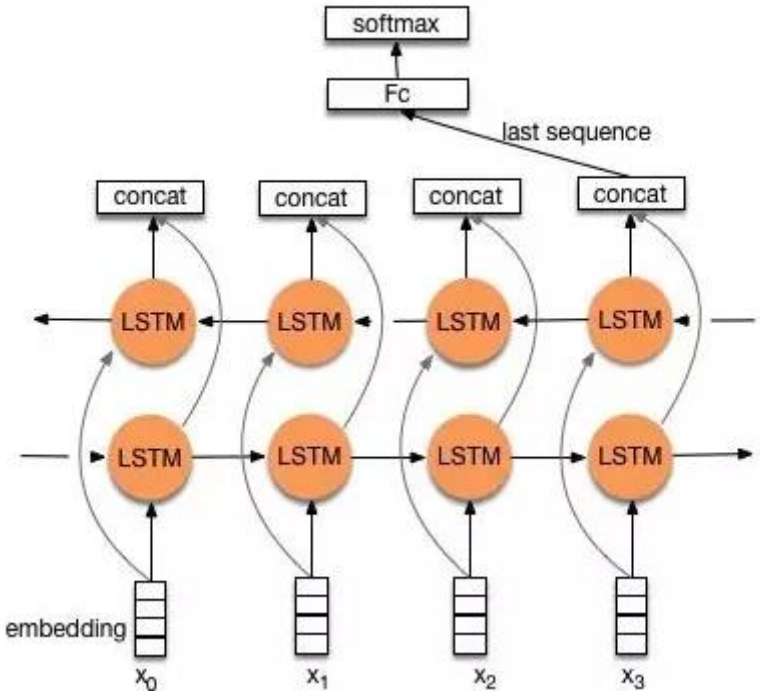

图 7 TextRNN 结构图

（2）Structure 2

流程：

$$embedding -> BiLSTM -> (dropout) -> concat\ ouput -> UniLSTM -> (droput) -> softmax\ layer$$

结构图如下图所示：

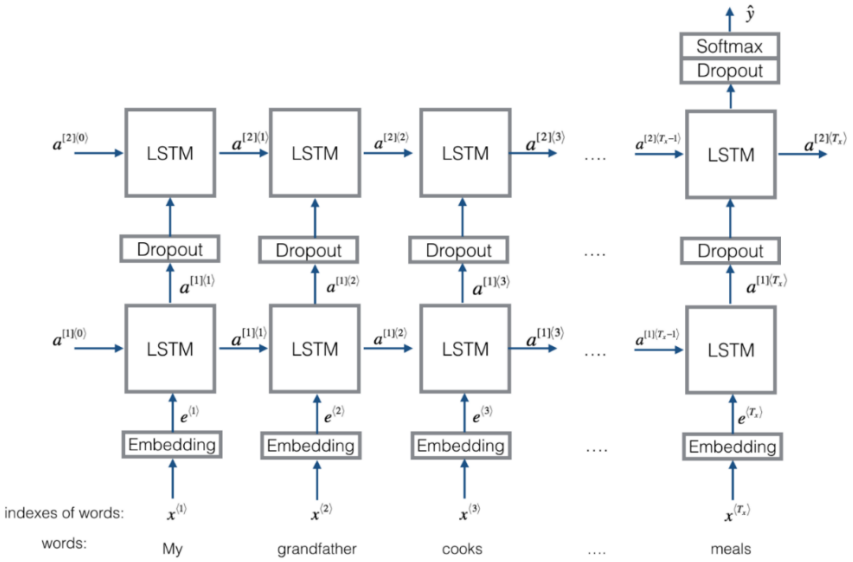

图 8 TextRNN 结构图

本题在选取 TextRNN 模型时，选用的第二种结构，设置隐藏层层数为 2，隐藏层

神经元为 128，$dropout$保留比例为 0.8. 把双向 LSTM 在每一个时间步长上的两个隐藏状态进行拼接，作为上层单向 LSTM 每一个时间步长上的一个输入，最后取上层单向 LSTM 最后一个时间步长上的隐藏状态，再经过一个 softmax 层(输出层使用 softamx 激活函数，2 分类的话则使用$sigmoid$)进行一个多分类。

3.处理过程

首先对其进行 Jieba 分词和去停用词。将分词好的中文词语通过 One-Hot 进行编码，通过 Word2Vec 模型中的 CBOW 模型这种 embadding 方式进行特征提取，将每个分词映射成一个词向量，获得初步词向量表示，将自然语言数值化，方便后续处理，对于每一个输入文本/序列，我们可以在 RNN 的每一个时间步长上输入文本中一个单词的向量表示，计算当前时间步长上的隐藏状态，然后用于当前时间步骤的输出以及传递给下一个时间步长并和下一个单词的词向量一起作为 RNN 单元输入，然后再计算下一个时间步长上 RNN 的隐藏状态，以此重复直到处理完输入文本中的每一个单词。

4.实验结果

通过进行 10 次 TextRNN 进行对留言内容分类验证，分别得到 10 组召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$的值，对其求取平均值则有：

表 4　TextRNN 四项指标

模型	召回率	精确度	准确率	$F1-score$
TextRNN	0.910	0.910	0.914	0.909

并得出其混淆矩阵：

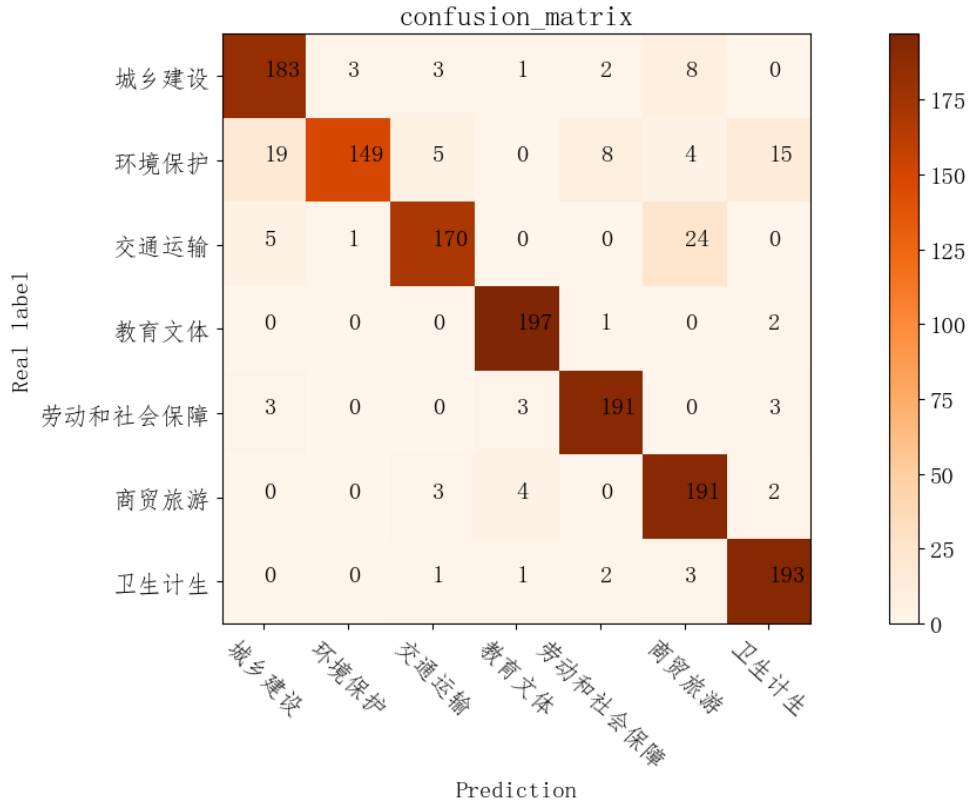

图 9 TextRNN 混淆矩阵

通过对基于 Word2Vec 的 LR 模型和 TextCNN 模型进行对比，可以看出：

表 5 基于 Word2Vec 的 LR 模型和 TextCNN 模型进行对比

模型	召回率	精确度	准确率	$F1-score$
LR	0.737	0.741	0.735	0.738
TextCNN	0.912	0.923	0.908	0.910
TextRNN	0.910	0.910	0.904	0.909

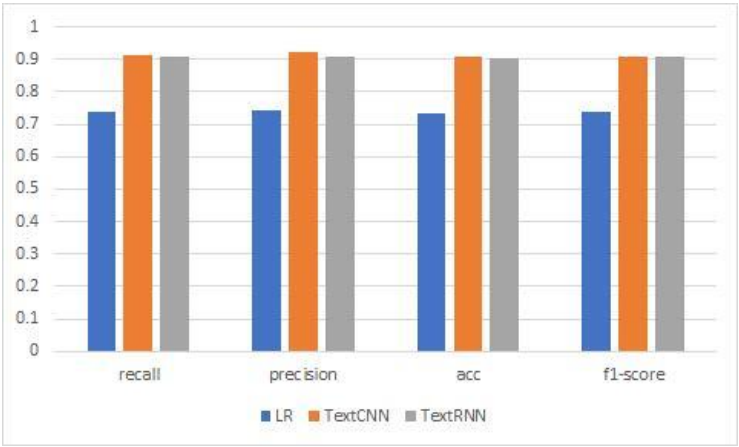

图 10 LR、TextCNN 和 TextRNN 评价指标对比图

3.1.4. 分类模型评价

（1）在以召回率（$Recall$）、精确度（$Precision$）、准确率（ACC）、$F1-score$为评价指标时，TextCNN 较逻辑回归模型的ACC提升了 17.3%,F1-score 提升了 17.2%,较 TextRNN 的ACC提升了 0.4%，F1-score 提升了 0.1%性能显著提高。基于 Word2Vec 的 TextCNN 模型相比于 LR 模型和 TextRNN 模型更具有优势，分类效果更好。

（2）对于训练时长可以得出，在基于此三种模型对题目给出数据进行训练时，通过多次训练，计算出 LR 模型平均训练时长为 13 分钟，TextCNN 模型训练平均时长为 20 分钟，而 TextRNN 训练时长平均为 1 小时 27 秒。

（3）在对模型交叉验证的过程中，笔者得出结果，对于模型的ACC值，通过分析结果的方差，得出，TextCNN 的方差较小。

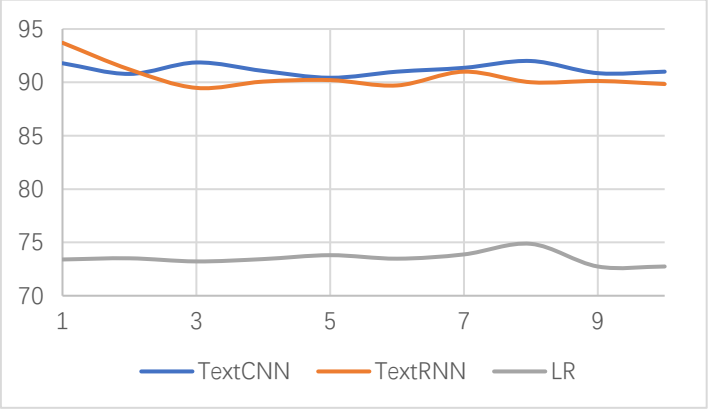

图 11 三种模型 ACC 率波动情况

结论：在针对训练效率、模型$F1-score$与交叉验证情况分析后，得出结论为：在基于文本内容进行一级标签分类时，优先选择 TextCNN 模型作为分类器。

3.2 热点问题挖掘

由题意可知，第二问要求对留言内容进行归类，定义热度评价指标，找到热点问题对其进行排名。现将对某一时间段内反映的特定地点或特定人群问题进行归类是我们首先要解决的问题。

3.2.1 对留言问题归类

依据题干，现将公众问政留言进行归类，由于数据量大，内容杂乱，主题不

一，很显然这是一个文本聚类问题，对其进行预处理成词向量后通过聚类模型DBSCAN进行聚类。

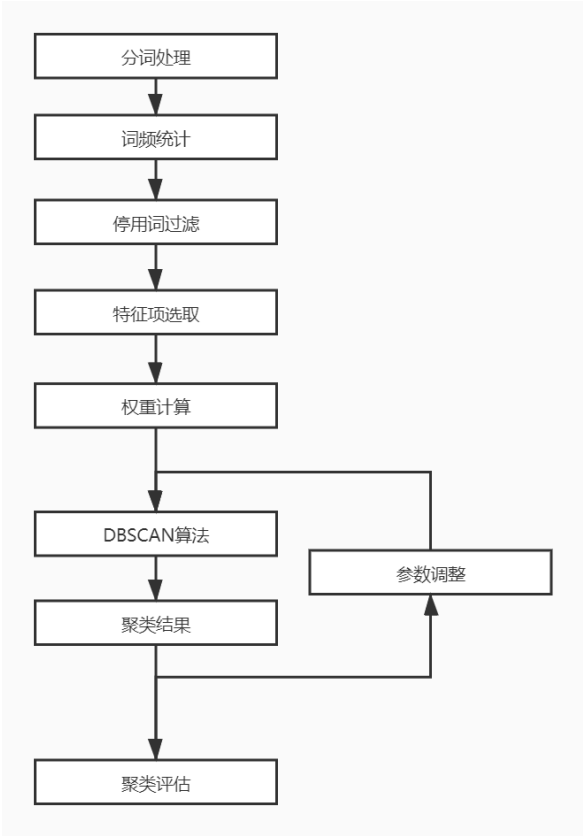

图12 文本聚类流程图

3.2.1.1 预处理

对于问政留言，由于文本较长，一般在100字以上，最多可达到数千字，可能会出现语句冗长、主题不明的问题。同问题一相似在进行预处理分词时，首先去掉留言中的一些虚词，比如：连词、代词、停用词等。接着对文本内容中出现的地名、组织名、电话号码、银行账号进行特殊处理。尽管这些词所代表的语义有限，但是出现频率可能很高，利用$TF-IDF$计算时，$TF-IDF$的值偏高，造成干扰，所以可能导致选择特征词项不准确，所以在计算文本相似度可以去掉他们。

1）jieba分词

分词是基于统计的分词方法，采用精确的分词模式，试图将句子最精确地切开，以用于文本分析。对训练集的两千余条样本数据循环遍历，使用Jieba库的cut方法获得分词列表赋值给变量。

2）基于词频统计的$TF-IDF$模型关键词抽取

$TF-IDF$算法是一种统计方法，是一种用于信息检索的加权技术，可以衡量一个字或一个文件的重要程度。TF-IDF 倾向于过滤掉常见词语，保留重要词

语。字词的重要性会随着其在文中出现的次数成正比增加，但是同时也会随着它在预料库中出现的频率成反比下降。简单来说其原理就是，如果某个单词在一篇文章中出现的频率 TF 高并且 IDF 的值越大说明该次在其他文章中很少出现，说明该词语对文章贡献率越大，我们就认为此词语有较好的类别区分能力，适合用来分类。

基于这一原理，我们对附件 3 中所给的留言文本集合 jieba 分词处理完的每一个分词进行$TF-IDF$算法处理，首先通过公式（12）对得到词频TF的值：

$$tf_{ij} = \frac{n_{ij}}{\Sigma_k n_{kj}} \qquad (12)$$

之后对 TF 的值进行归一化处理，让其短文本表示防止偏向长的文件：

$$TFIDF(w_i) = tf(w_i) \times idf(w_i)$$
$$= tf_j(w_i) \times lb(N/df(w_i)) \qquad (13)$$

这里假设某个词语w_i在留言文本j中出现的频率用$tf_j(w_i)$来表示，N是留言短信文本集合所包含的留言文本的条数，$df(w_i)$则是留言文本中包含某个词语w_i的总条数从而对每个词语进行分析得出其 TF-IDF 的值，并对每一条留言文本中词语的$TF-IDF$的值进行排序，筛选。通过给定的阈值m（$0 < m < 1$），这里的阈值$m = 0.5$,从$TF-IDF$高值到低值，选择排序的实词w_i使得 $TFIDF(w_i) > m$,从而将这些实词w_i作为留言文本中的特征词，构成了留言文本的特征向量，降低了模型的维度。

3.2.1.2 余弦相似度计算

使用 Word2Vec 对词汇集进行训练，得到词向量，对每一个词进行K维词向量表征，然后通过计算余弦夹角，得到每个词和其他词汇之间的相似度：

$$Sim(e_i, f_j) = cos\,\theta = \frac{e_i \cdot f_j}{\|e_i\| \cdot \|f_j\|} \qquad (14)$$

其中e_i是源文档句子中的第i个词，f_j是目标文档句子中的第j个词，第i个词与第j个词之间的相似度为$Sim(e_i, f_j)$，而e_i, f_j为词向量表示。

3.2.1.3 DBSCAN 聚类算法

DBSCAN 聚类算法是一种基于密度的聚类算法，这类算法一般假定类别可以通过样本分布的紧密程度决定。同一类别的样本，他们之间是紧密联系的。通过将紧密联系的样本划为一类就得到了一个聚类类别。通过将所有各组紧密联系的样本划分为各个不同的类别，则我们就得到了最终的所有聚类类别结果。本题通过 DBSCAN 聚类算法迭代 100 次对预处理的留言文本进行聚类。

原理：DBSCAN 需要二个参数：扫描半径 (eps)和最小包含点数 $(minpoints)$。本题进行实验中扫描半径 (eps)为 0.25，最小包含点数 $(minpoints)$为 5。任选一个未被访问$(unvisited)$的点开始，找出与其距离在 eps之内(包括eps)的所有附近点。如果 $附近点的数量 \geq minpoints$，则当前点与其附近点形成一个簇，并且出发点被标记为已访问$(visited)$。 然后递归，以相同的方法处理该簇内所有未被标记为已访问$(visited)$的点，从而对簇进行扩展。如果 $附近点的数量 < minpoints$，则该点暂时被标记作为噪声点。如果簇充分地被扩展，即簇内的所有点被标记为已访问，然后用同样的算法去处理未被访问的点。如图所示：

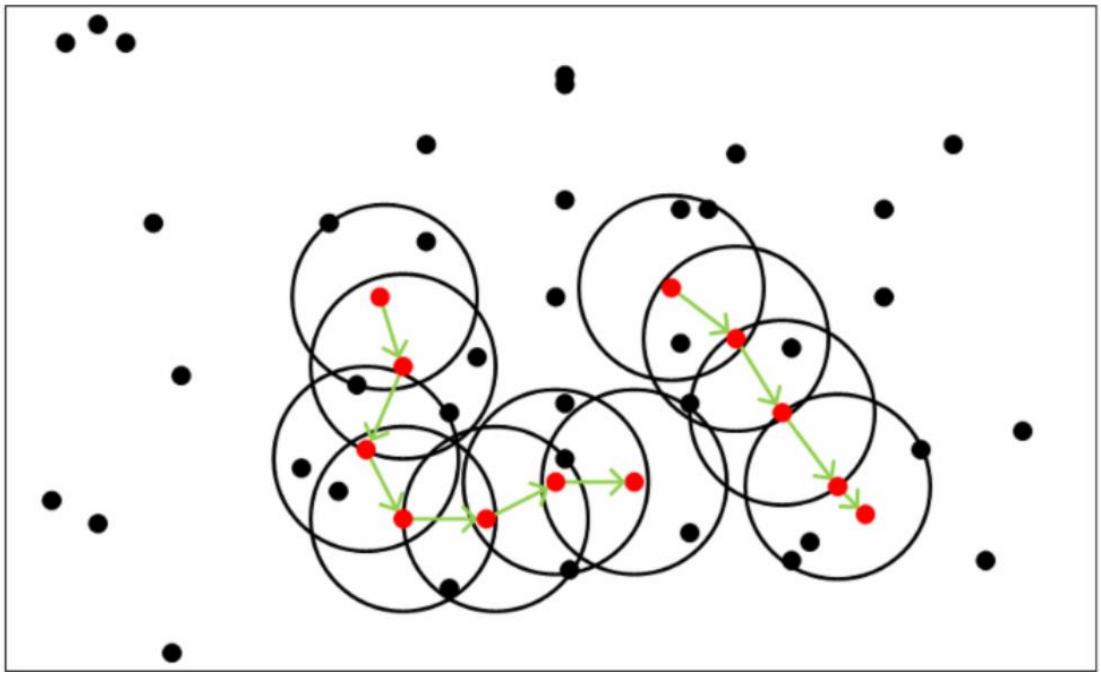

图 13　DBSCAN 聚类算法

3.2.1.4 互信息进行聚类指标评价

互信息是用来衡量两个数据分布的吻合程度。假设U与V是对N个标签的分配情况，则两种分布的熵分别为：

$$H(u) = \sum_{i=1}^{|U|} P(i)\, log\big(P(i)\big) \qquad （15）$$

$$H(V) = \sum_{j=1}^{|V|} P'(j)\, log\big(P'(j)\big) \qquad （16）$$

其中$P(i) = |U_i| \,/\, N$，$P'(j) = |V_i| \,/\, N$。U与V之间的互信息（MI）定义为：

$$MI(U,V) = \sum_{i=1}^{|U|} \sum_{j=1}^{|V|} P(i,j)\, log\left(\frac{P(i,j)}{P(i)P'(j)}\right) \qquad （17）$$

其中$P(i,j) = |U_i \cap V_i| \,/\, N$。标准化后的互信息为：

$$NMI(U,V) = \frac{MI(U,V)}{\sqrt{H(U)H(V)}} \qquad （18）$$

调整互信息定义为：

$$AMI = \frac{MI - E|MI|}{max\big(H(U),H(V)\big) - E|MI|} \qquad （19）。$$

利用基于互信息的方法来衡量聚类效果，MI与NMI取值范围是$[0,1]$，AMI取值范围是$[-1,1]$，他们都是值越大意味着聚类越符合。本题中通过计算得出

$$AMI = 0.805$$

结果较高，说明聚类效果较好。

3.2.2 定义热度评价指标

为了更好的定义留言热点问题的热度评价指标，我们首先将留言文本定义为一个 7 元组W：

W=（$number$, $user$, $subject$, $text$, $time$, $attitude\_num$, $against\_num$）其中各变量表示为：

表 6　各元组量的含义

元组	含义
$W.user$	表示该留言的发表用户
$W.number$	表示该留言的编号
$W.time$	表示该留言的时间
$W.text$	表示该留言的详细内容
$W.attitude\_num$	表示该留言的点赞数
$W.subject$	表示该留言的主题
$W.against\_num$	表示该留言的反对数

考虑一条留言是否反映的是热点话题，则需要从留言出现的次数、对留言的点赞数以及反对数三个层面进行热度的刻画。这三个因子表示了市民对此问题的关注度以及市民的主观情感。因此，我们在本题中选取留言h在时间t内的相关留言问题的出现次数$W_count = W_t^h$、平均点赞数 $attitude\_avg$、平均反对数平均点赞数 $against_avg$作为衡量热点问题$topic_hot$指标。各指标的计算公式如下：

$$w\_count = W_t^h \quad (20)$$

$$attitude\_avg = \frac{\Sigma w.attitude_{num}}{W_t^h} \quad (21),$$

$$against\_avg = \frac{\Sigma w.against_num}{W_t^h} \quad (22) \quad (w \in W_t^h)$$

W_t^h表示在时间间隔t内留言h的讨论量。由各个指标进行加权求和：

$$topic\_hot = \beta_1 W\_count + \beta_2 attitude\_avg + \beta_3 against\_avg \quad (23)$$

求出$topic_hot$的值进行比较得出前五名的问题即为热点问题：

表 7 排名前五热点问题

热度排名	问题 ID	热度指数	时间范围	问题描述
1	1	90947.7459	2019.7.2-2019.9.1	A 市伊景园滨河苑捆绑销售车位
2	2	59513.33333	2019.11.2-2020.1.26	A 市丽发新城小区附近搅拌站噪音扰民和污染环境
3	3	24678.88713	2019.1.8-2019.7.8	严惩 A 市 58 车贷特大集资诈骗案保护伞
4	4	4906.795132	2019.1.16-2019.9.27	咨询 A 市人才购房补贴政策
5	5	4685.897436	2019.2.16-2019.11.12	A 市江山帝景新房脏乱差，有安全隐患

由此可以找出排名前五的热点问题，详细见文件“热点问题表.xls”和热点问题对应的留言信息“热点问题留言明细表.xls”.

3.3 答复意见的评价

本题要求针对附件 4 相关部门对留言的答复意见，我们从内容出发，进行对答复意见的质量给予一套评价方案。

首先提取影响留言答复意见的评价指标，然后构建答复意见质量的评价指标体系与模型，最后验证模型性能。具体流程如图所示：

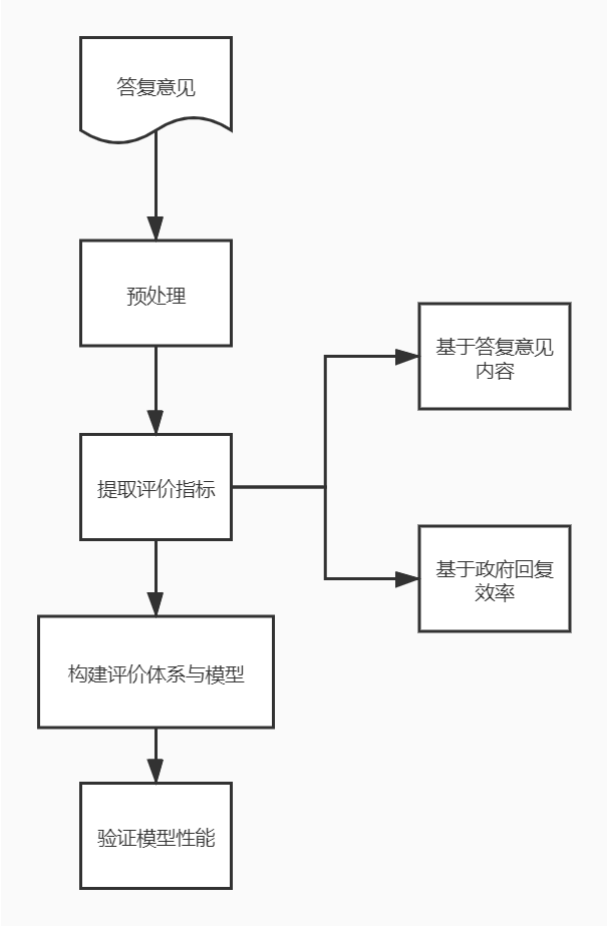

图 14 答复意见评价流程图

3.3.1 预处理

同第一问一样，我们仍然采用 Jieba 分词和 Word2Vec 模型将答复意见的文本内容用向量表示。所以首先对其进行 Jieba 分词和去停用词。将分词好的中文词语通过 One-Hot 进行编码，通过 Word2Vec 模型中的 CBOW 模型将其映射到数值空间，进行特征提取，获得初步词向量表示。

3.3.2 提取评价指标

指标提取是对答复意见进行分析的另一项重要任务。在这里我们选取的指标除了题目中给出的相关性、完整性、可解释性外，还选择了信息量、一致性、可信性。

基于文本内容：

（1）完整性（*Complete*）：

可以理解为一条答复评论既包含的数值型评论也包含文本型评论，简单来说，完整的数据可以标注为 1，不完整则标记为 0；

（2）一致性（$Coherent$）：

可以简单地认为评论内容所表达的情感与打分是否一致；本题中，我们将答复的一致性得分记为Co，文本型评论极性得分记为P_1，数值型评论（评分）得分记为P_2，如果两种评论一致（即$P_1 = P_2$）时，则一致性得分$Co = 1$，不一致则$Co = 0.5$.

（3）可信性（$Credibility$）：

表示某回复是否可信或有效。 通过检索答复意见中是否出现引用，如：法律，文件等。

（4）可解释型（$Readability$）：

可解释性对与中文答复意见没有多大意义，本题我们将其理解为可读性，通过自动化可读性指数 ARI 来表示，ARI依赖于文本的字符数，比其他的可读性测试方法准确率高，ARI具体公式为：

$$ARI = 4.71 * \left(\frac{总字符数}{总字数}\right) + 0.5 * \left(\frac{总字数}{总句数}\right) - 21.43 \quad （23）$$

其近似值等于我们可能理解一段文字的最低程度。

（5）相关度（$Relevancy$）：

基于词频统计的$TF - IDF$模型对关键词抽取，给定阈值$m = 0.5$，求出$TFIDF(w_i)$的值，当$TFIDF(w_i) > m$，从而将这些实词w_i作为留言文本和答复意见中的特征词，构成了留言和答复意见的特征向量，降低了模型的维度。之后通过余弦相似度的计算，得出每个答复意见中的特征词和留言特征词之间的相似度，从而反应文本相似度 。

（6）信息量（$Words$）：答复意见的长度。

（7）基于答复工作效率($Efficiency$)：

由于附件 4 中的数据中给出市民留言时间和政府答复时间，且考虑到处理问题的时效性，所以我队从时间角度，对政府回复的时间 进行处理，衡量政府对于市民留言的问题处理的及时性以及政府的办公效率。

3.3.3 构建评价体系与模型

根据所选取的指标，我们将答复意见评价指标体系定义为$1W2R3C1E$指标，用广义线性回归方法（GLM）构建答复意见评价模型。在此用 Quality 表示答复意见质量，得分记为 Q，可建立如下回归模型：

$$\begin{aligned} Q = {} & w_1 Word + w_2 Relevancy + w_3 Readbility \\ & + w_4 Complete + w_5 Coherent + w_6 Credibility \\ & + w_7 \text{Efficiency} + \varepsilon \qquad (24) \end{aligned}$$

其中，ε表示常数项，w_i=（w_1，w_2，w_3，w_4，w_5，w_6，w_7）表示各指标对应的权重，本题中，我们将权重设为均权对Q进行求解，实验得出的数据Q都是在0~100之间的数。同时给出一套评价指标进行等级分类，如：100~80 *非常满意*、80~60 *满意*、60~40 *良好*、40~20 *中等*、20~10 *不满意*。来评价政府部门的答复意见的质量。

四．实验评价

本节我们对上文中建立的模型进行评价，通过传统方法和深度学习的方法进行比较，从而说明我们的模型的合理性与有效性，但是也是仍然存在不足。

4.1 模型评价

问题一，LR 模型优点预测结果是界于 0 和 1 之间的概率；可以适用于连续性和类别性自变量；容易使用和解释；缺点：对模型中自变量多重共线性较为敏感；很多区间的变量变化对目标概率的影响没有区分度，无法确定阈值 TextCNN 模型优点：准确率，精确率较高，运行速度快。缺点可能会出现过拟合现象。TextRNN 准确率，精确率较高，但是运行速度太慢。总体说，对于问题一需要更多的数据进行模型分析，但是由于时间和机器配置和授权问题，无法在规定的时间内寻找更多的数据进行训练。但是分别进行了 10 交叉验证，准确率较高。

对于模型训练，由于附件给出的数据类别不平衡，为平衡数据，笔者在青岛政务网（http://zxwz.qingdao.gov.cn/Site/ListQue.aspx）自行爬取 4000 余条数据，添加到原训练集进行模型训练后准确率有明显提高，但是，由于爬取数据与题目提供数据未出自同一地区，结果出现了过拟合现象，针对此类问题可以在本模型训练的过程中，对数据量进行加大，并对数据的平衡化进行处理，这样可以在本模型的基础上提高部分准确率。

问题二，选用基于词频的$TF-IDF$进行特征词向量提取，准确性较高，通过 DBSCAN 聚类算法进行聚类，通过加权处理定义热点问题的热度评价指标准确性高。

问题三，首先提取影响留言答复意见的评价指标，然后构建答复意见质量的评价指标体系与模型，最后验证模型性能。提取指标时，分别从基于答复意见内容和答复时间两个方面进行考虑，比较全面客观，并对聚类模型进行评价。

4.2 实验平台

实验环境的软件硬件配置如表所示：

表 8 实验环境配置

处理器	Inter Core i7-9750H
RAM	16G
显卡	GTX1660Ti MaxQ
操作系统	Windows 10
python	3.5.2
tensorflow	1.2,1
gensim	2.3.0
Scikit-learn	0.19.0

参考文献

[1]梁柯,李健,陈颖雪,刘志钢.基于朴素贝叶斯的文本情感分类及实现[J].智能计算机与应用,2019,9(05):150-153+157.
[2]谭鹏,罗顺莲,孙小淞,王惠,梁晓菡.基于小波神经网络的话题热度预测模型研究[J].现代信息科技,2018,2(05):74-78.
[3]宁建飞,刘降珍.融合 Word2Vec 与 TextRank 的关键词抽取研究[J].现代图书情报技术,2016(06):20-27.
[4]陈曦.文本挖掘技术在社情民意调查中的应用[J].中国统计,2019(06):27-29.
[6]涂文博,袁贞明,俞凯.针对文本分类的神经网络模型[J].计算机系统应用,2019,28(07):145-150.
[7]郭银灵. 基于文本分析的在线评论质量评价模型研究[D].内蒙古大学,2017.
[8]徐旖旎. 基于微博的媒体奇观网络舆情热度趋势分析[J]. 情报科学, 2017, 035(002):92-97,125.
[9]齐向明, 孙煦骄. 基于语义簇的中文文本聚类算法[J]. 吉林大学学报(理学版), 2019, 057(005):1193-1199.
[10]盛明科, 刘贵忠. 政府服务的公众满意度测评模型与方法研究[J]. 湖南社会科学, 2006(06):41-45.
[11]王庆龙. 基于 word2vec 和 SVM 的文本内容监测分析应用研究[D].
[12]安波. 基于逻辑回归模型的垃圾邮件过滤系统的研究[D]. 哈尔滨工程大学, 2009.
[13]吴柳, 程恺, 胡琪. 基于文本挖掘的论坛热点问题时变分析[J]. 软件, 2017, 038(004):47-51.
[14]刘金岭, 宋连友, 范玉虹. 基于语义信息的中文短信文本相似度研究[J].计算机工程, 2012, 38(13):58-60.
[15] https://github.com/fxsjy/jieba
[16] www.tensorflow.org/
[17] https://github.com/ownthink/Jiagu
[18] https://github.com/tsroten/pynlpir
[19] http://zxwz.qingdao.gov.cn/Site/ListQue.aspx