“智慧政务”中的文本挖掘应用

摘 要

本文利用自然语言处理技术和文本挖掘的方法对互联网公开来源的某群众问政留言数据进行分析，并针对具体问题建立了相应数学模型，为提升政府的管理水平和施政效率提供一定的参考价值。

在数据预处理阶段，本文从数据源文件中，抽取留言文本进行分词和去除停用词处理，并剔除了特殊字符和异常数据。通过 Word2vec 获得文本的词向量表示。为了加大文本数据的差异性，避免模型过拟合，本文使用了 shuffle、drop、同义词替换、文本裁剪等方法进行数据增强，来提高模型的泛化能力，突出文本的特征。

针对问题一：建立了多标签文本分类模型。其中分别采用了传统机器学习中的朴素贝叶斯分类器和 TextRNN、TextCNN、FastText 和 Bert 四类深度学习文本分类模型进行留言内容的分类。为了提高模型的分类效果，本文将四种深度学习模型的分类结果通过线性加权的方式进行融合，显著地提升了模型的得分。随机选取了 210 条数据作为测试集，得到 F1 值如下表：

模型	贝叶斯	TextRNN	TextCNN	FastText	Bert	融合模型
F1Score	0.8350	0.9064	0.9052	0.8971	0.9152	0.9243

由上表可知，采用深度学习的方法进行文本分类，效果明显好于传统的机器学习方法。同时在进行模型融合时发现：模型和数据差异越大，融合时对得分的提高更加显著。

针对问题二：为了挖掘群众留言中的热点问题，本文使用 TF-IDF 算法提取出了留言文本的特征向量，然后对得到的特征向量使用 DBSCAN 算法进行留言分类，并使用改进的 Reddit 热度排名算法计算出每组热点问题的平均热度值，求解出排名前 5 的热点问题。最后使用 TextRank 算法从每类留言中提取出影响力最大的留言标题作为热点问题，并得到了相应热点问题对应留言明细表。

针对问题三：建立基于 BP 神经网络的线性加权综合评价模型。本文从答复的相关性、完整性、可解释性和时效性四个方面对答复意见进行量化评估。根据附件 4 中已有的数据，计算出每条答复意见的四维指标数值，然后将留言详情和答复意见作为输入层，以答复的相关性、完整性、可解释性和时效性四维指标为输出层，使用 BP 神经网络进行训练，最后使用四维指标线性加权作为答复意见的综合评价值，求解给出排名前 10 的答复意见。

关键词：多标签文本分类　DBSCAN　TextRank　BP 神经网络

Abstract

This article uses natural language processing technology and text mining methods to analyze the data of a certain public's public opinion on the Internet, and establishes a corresponding mathematical model for specific problems, which provides a certain reference value for improving the government's management level and governance efficiency.

In the data preprocessing stage, this article extracts the message text from the data source file for word segmentation and removal of stop words, and removes special characters and abnormal data. Get the word vector representation of the text through Word2vec. In order to increase the difference of text data and avoid overfitting of the model, this paper uses methods such as shuffle, drop, synonym replacement, and text clipping to enhance the data to improve the generalization ability of the model and highlight the characteristics of the text.

Aiming at Problem 1: A multi-label text classification model is established. Among them, the naive Bayes classifier in traditional machine learning and the four deep learning text classification models of TextRNN, TextCNN, FastText and Bert are used to classify the message content. In order to improve the classification effect of the model, this paper combines the classification results of the four deep learning models through linear weighting, which significantly improves the model's score. Randomly selected 210 data as the test set, and obtained the F1 value as follows:

Model	Bayes	TextRNN	TextCNN	FastText	Bert	Fusion model
F1Score	0.8350	0.9064	0.9052	0.8971	0.9152	0.9243

It can be seen from the above table that the text classification using the method of deep learning is significantly better than the traditional machine learning method. At the same time, it was found during model fusion that the greater the difference between the model and the data, the more significant the improvement in score during fusion.

Aiming at Problem 2: In order to mine the hot issues in the mass messages, this paper uses the TF-IDF algorithm to extract the feature vector of the message text, and then uses the DBSCAN algorithm to classify the message, and uses the improved Reddit ranking algorithm The average heat value of each group of hotspot problems is found to solve the top 5 hotspot problems. Finally, the TextRank algorithm is used to extract the most influential message title from each type of message as a hotspot issue, and the corresponding hotspot issue corresponding message list is obtained.

Aiming at Problem 3: Establish a linear weighted comprehensive evaluation model based on BP neural network. This article quantitatively evaluates the opinions of the answers from four aspects: relevance, completeness, interpretability and timeliness. Based on the existing data in Annex 4, calculate the four-dimensional index value of each reply opinion, then use the message details and reply opinions as the input layer, and output the four-dimensional indicators of relevance, completeness, interpretability and timeliness of the reply as the output Layer, use BP neural network for training, and finally use four-dimensional index linear weighting as the comprehensive evaluation value of the reply opinion, and solve to give the top 10 reply opinions.

Keywords: Multi-label text classification DBSCAN TextRank BP neural network

目录

一．问题重述

1.1 问题的相关背景

随着物联网、云计算、移动互联网等的发展，“智慧政务”成为政府办公不可或缺的一部分。“智慧政务”服务包含医疗、交通、教育、缴费等多方面的民生服务办事功能，实现了“信息多跑路，群众少跑路”。

“智慧政务”可以让领导在线上或得民生问题，第一时间就能看到群众提交的诉求，能更快更好的解决问题。在智能服务方面，能够感知和预测群众所需的服务，并为其提供个性化的服务。“智慧政务”提高了政府办公、监督、决策和服务的智能化水平，形成了敏捷、便民、高校的新型政府。

1.2 题目的相关信息

题目中给出了互联网公开的群众问政留言记录，并给出了政府相关部门对部分群众留言的答复意见。根据题目所给信息，解决以下问题：

1.3 需要解决的问题

问题一：建立关于留言内容的一级标签分类模型，对留言的一级文本分类。

问题二：建立模型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，并给出排名前 5 的热点问题和热点问题留言明细。

问题三：多角度对答复意见的质量给出一套评价方案。

二．符号说明

符号	符号说明
$P(s_i \mid x)$	表示待分类文本 x 属于类别 s_i 的概率
P	分类器预测为正且预测正确的样本占所有预测为正的样本的比例
R	分类器预测为正且预测正确的样本占所有真实为正的样本的比例
TF_{ij}	表示第 i 个关键词在第 j 个文档中出现的频率
β	表示留言的点赞数和反对数之间差的绝对值
f	表示阻尼系数，即按照超链接进行浏览的概率
$X(k,Q)$	表示第 k 个留言详情中关键字及其权重向量 Q
$X(k,A)$	表示第 k 个答复意见中关键字及其权重向量 A
ρ_i	表示第 i 个答复文本完整性的度量
l_{ik}	表示第 i 条答复意见的词向量长度
l_{ij}	表示第 i 条留言详情的词向量长度
n_{ip}	表示第 i 条答复意见的第 p 个推测性词汇

注：表中未列出符号及重复的符号以文章出现处为主。

三．问题分析

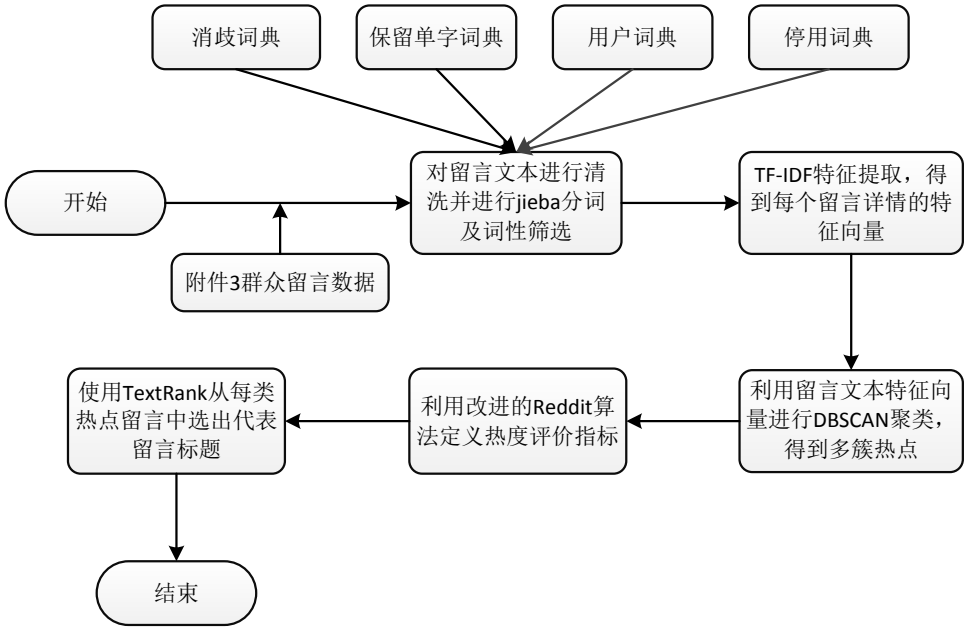
本文要求根据网络问政平台的留言记录和相关部门对部分群众留言的答复意见进行文本挖掘和处理。但是有群众留言的存在着差异性和无规则性，大量的留言文本不能直接用来建模分析。因此，本文对原始留言文本进行数据预处理：特殊字符处理、分词和词性处理、去停用词和 word2vec 向量表示处理。同时由于文本条数和强度不够，因此本文使用 shuffle 和 drop 两种方法进行数据增强处理。最后对处理后的数据进行建模分析。

3.1 问题一的分析

问题一要求对留言的一级文本分类，而附件一给出留言的三级分类情况，因此需要根据这些标签建立分类模型。根据数据预处理得到字词数据，本文从传统机器学习和深度学习两个方面建模进行对比分析选取精度最高的模型。在机器学习方面使用朴素贝叶斯分类模型，在深度学习方面使用 Text-RNN、TextCNN、FastText、Bert 四种分类模型，然后从深度学习的四种模型中选择模型进行概率等权重融合建立新的多分类模型，最后选择最高精度模型作为该问的分类模型。

3.2 问题二的分析

问题二要求根据附件 3 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，并给出排名前 5 的热点问题和热点问题留言明细。因此首先需要将所有的留言进行特征选择，挑选出时间、地点和问题三个要素，然后将所有的留言问题进行分类，并利用改进的 reddit 算法对留言时间和留言投票情况进行热度评价指标的量化，最后根据热点排名对进行留言热点主题的提取，得到留言明细情况，具体的分析和处理流程图如下：



3.3 问题三的分析

问题三要求对答复意见的质量给出一套评价方案，因此本文根据留言详情和答复意见，确定答复意见的相关性、完整性、可解释性和时效性等四个评价指标，具体的量化过程详见问题三的模型建立部分。然后建立 BP 神经网络评价模型，以留言详情和答复意见作为输入，四个评价指标作为输出，这样就能得到每条答复意见的四个评价指标的评价值，最后通过线性加权得到每条答复意见的综合评价值，本文给出答复意见质量排名前 10 的留言情况。

四．数据处理

4.1 数据预处理

本文主要是对群众问政留言进行文本挖掘，在建模求解之前进行数据预处理。由于所给的群众留言文本中的字词较多，而且存在许多无意义的字词和特殊字符，计算机无法直接进行识别和计算。因此，首先针对这些问题进行处理，然后使用 word2vec[1]进行文本向量表示，为后续研究做准备。整体流程如下：

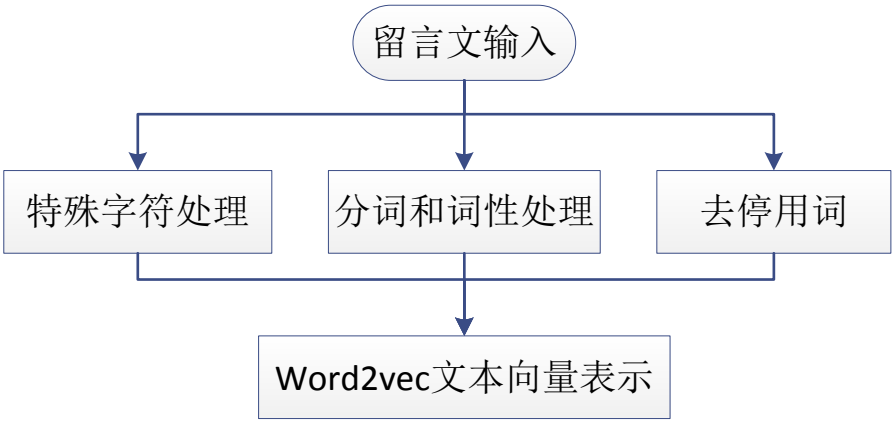


图 1：数据预处理流程图

(1).特殊字符处理

对于特殊字符的处理，在本题的数据处理中，因为主要是针对群众留言内容处理和分析，在留言内容的分析过程中，我们发现在留言内容中，有许多无关的符号和数字表情，如“###”、“！！！”等差异化问题，计算机无法识别的符号和数字表情应该进行相应处理。因此，在关于特殊字符的处理采用正则化表达式进行相关的处理和分析过程。

(2).分词和词性处理

由于中文文本的特点是词与词之间没有明显的界限，从群众留言文本中提取词语时需要分词和词性处理，本文采用 Python 开发的一个中文分词模块——jieba 分词，对留言主题和留言详情中的每一句话分词和词性处理。

(3).去停用词

在文本处理中，停用词是指那些功能作用极其普遍，与其他词相比没有什么实际意义的词，它们通常是一些单字，单字母以及高频的单词，比如中文中的“的、了、地、吗，哈”等，英文中的“the、 this、 an、 a、 of”等。对于停用词一

般在预处理阶段就将其删除，避免对文本，特别是短文本，造成负面影响。此处本文选用效果较好的哈工大中文停用词表。

(4). word2vec 文本向量表示

为了将语料文本能输入神经网络进行训练，首先要将自然语言符号表示成计算机能够理解的数字形式，也就是把每个词表示为一个很长的向量。本文使用word2vec用高维的向量来表示词语，并把具有相近意思的词语放在相近的位置，且固定词向量的维度，就可以通过实数向量来训练模型，以此获词语的词向量表示。具体来说，Word2vec 中涉及到了两种算法，一个是 CBOW，一个是 Skip-Gram。其结构示意图如下：

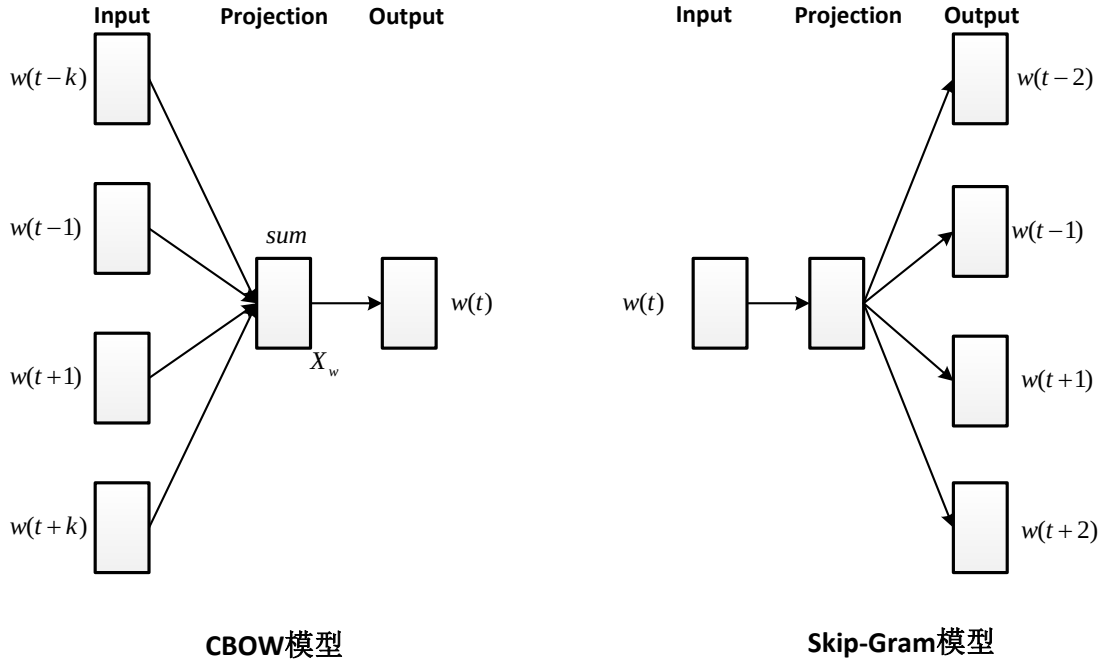


图 2：Word2vec 算法网络示意图

本文使用 Skip-gram 模型构建群众留言留言分类系统。Skip-gram 模型的训练目标就是使得下式的值最大：

$$\frac{1}{N}\sum_{t=1}^{N}\sum_{-e\le i\le e}\log p(w_{t+i}\mid w_t)$$

其中，e 表示窗口大小，N 是训练文本大小。基本的 Skip-gram 模型计算条件概率如下式：

$$p(w_o\mid w_i)=\frac{\exp(v_{wo}\bullet v_{wi})}{\sum_{w=j}^{W}\exp(v_{wo}\bullet v_{wi})}$$

其中，v_{wi} 和 v_{wo} 分别表示的词语 w 的输入和输出向量，W 是词典的大小。

4. 2 数据增强

由于提供的本文数据中，大部分都是相同的格式，导致数据之间的特征不明显，容易导致训练的过拟合。因此我们采用一些数据增强的方法，提高模型的泛化能力，突出文本的特征。

4.2.1 随机 drop 和 shuffle

shuffle 是指对留言文本内容进行随机打乱，随机 drop 是对留言内容进行随机删除部分，避免文本格式相似度太高从而导致训练过拟合。具体处理过程如下图：

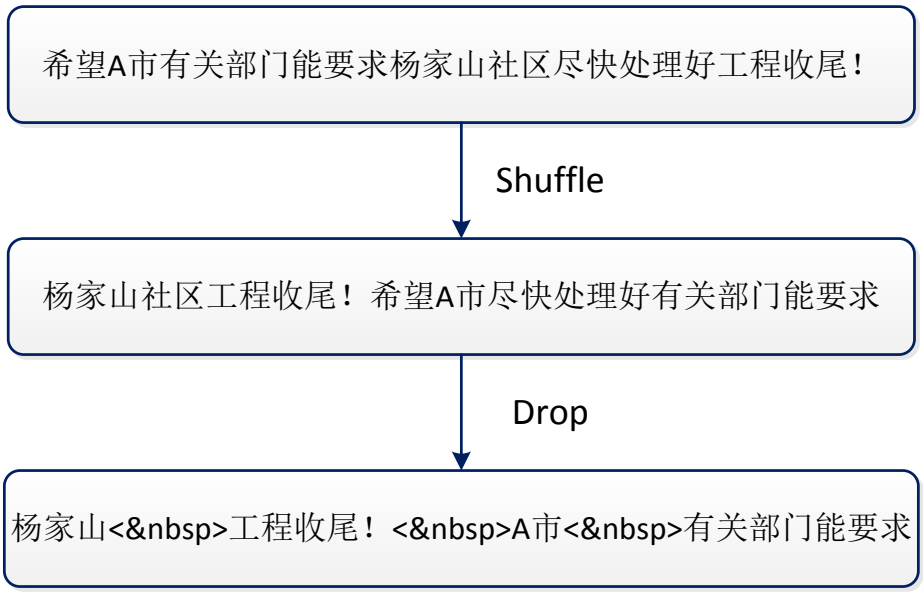


图 3：处理过程图

4.2.2 同义词替换

在一些句子当中，同一个意思可能出现多种不同的词语表达，可以随机的选一些词并用它们的同义词来替换这些词，这样句子仍具有相同的含义，很有可能具有相同的标签。但这种方法对我的任务来说没什么用，因为同义词具有非常相似的词向量，因此模型会将这两个句子当作相同的句子，而在实际上并没有对数据集进行扩充。具体处理流程如下：

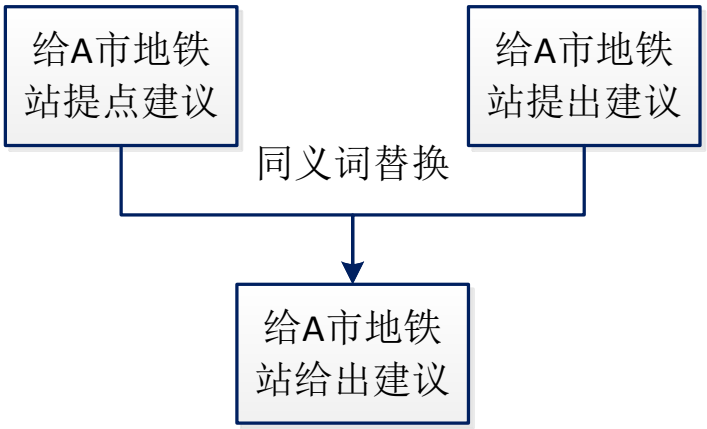


图 4：处理流程图

4.2.3 文本裁剪

大部分留言的长度都不相同，部分留言存在很多无意义的语义，因此对于不同长度的问题文本，为了方便模型训练，本文将所有的留言文本截取成相同的长度（文本长度为 128）。太短的就补空格，太长的就截断。操作图示如下：

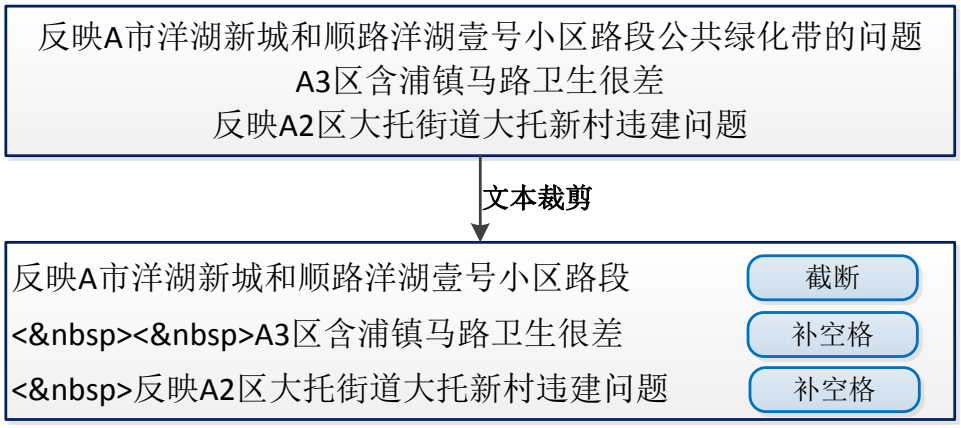


图 5：操作图示

4. 3 数据分析

4.3.1 附件 2 群众留言标签分类统计

通过对附件 2 中的数据进行标签统计，得到标签个数分布如下图所示：

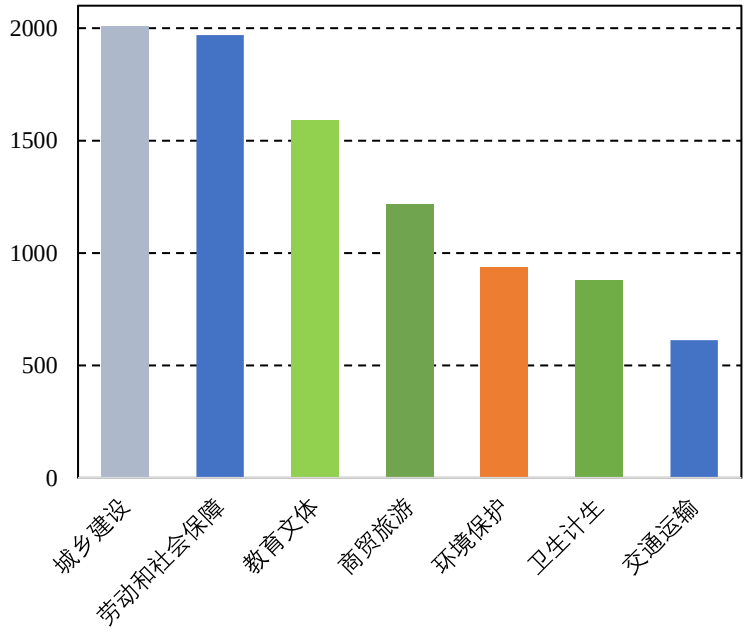


图 6：标签个数分布

由上图可知，群众留言中城乡建设和劳动社会保障问题居多，交通建设方面的群众留言最少。这说明该互联网问政平台中，可能更多的存在对中下层劳动人民的利益发展问题，而道路交通建设方面较好。因此，推测群众反映的热点问题可能主要集中在城乡建设和劳动社会保障问题方面。

4.3.2 群众留言的训练集和测试集长度分布

本文将附件 2 中的 9210 条数据进行随机打乱，划分成了训练集和测试集，其中训练集和测试集的留言标题长度和内容长度分布如下图所示：

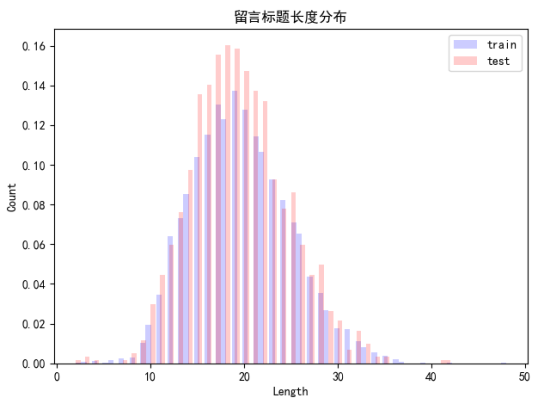


图 7：留言标题长度图

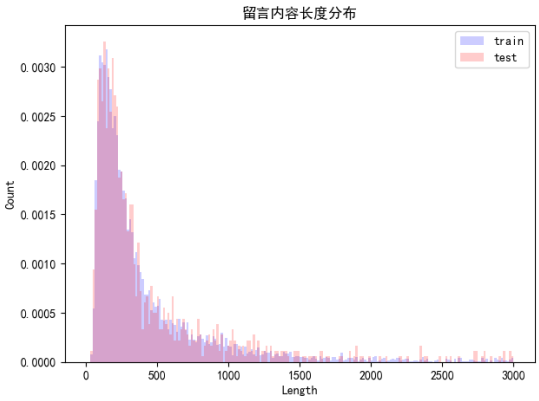


图 8：内容长度分布图

根据上图可以看出留言的标题长度符合正态分布，作为训练训练较为合理，但是留言内容中有部分长度超过 1000 个字符，长度较大，因此在训练之前利用 pad_sequence 函数进行一定的文本长度截取，避免部分数据长度过大影响模型的整体效果。同时由图中训练集和测试集的比例可以看出，我们在选取训练数据和测试数据的时的长度比例基本吻合，说明测试数据对整体数据有一定的代表性。

4.3.3 附件 3 群众留言的赞成数和反对数统计

本文对附件 3 所有的留言的点赞和反对的情况进行统计，在统计之前删去赞成和反对数为 0 的留言，得到的区间分布结果如下图：

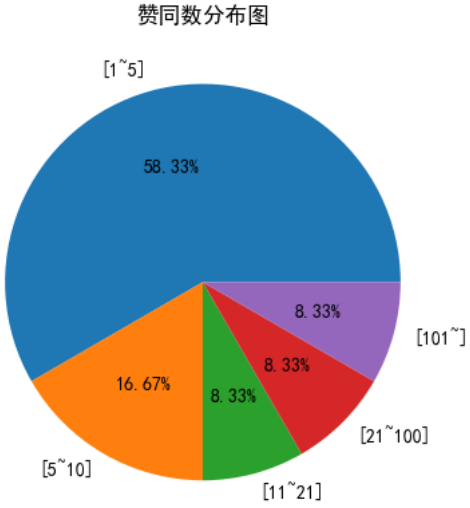


图 9：赞成数分布图

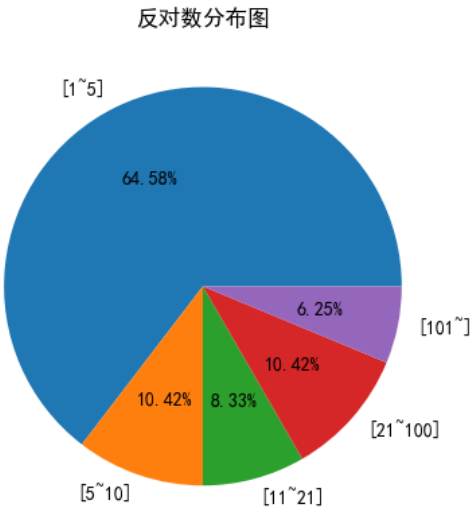


图 10：反对数分布图

由于大部分的留言中，赞成数和反对数为 0 的较多，因此这里进行去 0 处理再统计。从上图可知，群众留言的赞成和反对的票数区间占比情况很接近，这说明某一留言得到其他人的赞成和反对是具有可信性的。赞成数和反对数主要集中在[1~5]区间，在票数高达 101 的占比较少。这也反映出，群众留言是会得到其他群众的赞成和反对，是会得到其他群众的反映。

五．群众留言分类

问题一要求根据群众的留言内容进行以及标签的分类，即对留言主题和留言

详情进行分析。因此，本文在这里使用传统机器学习中的贝叶斯分类模型与深度学习中常见的四个文本分类模型进行对比，通过 F-score 对这些模型的精度进行对比分析，由于这些模型都的应用已经十分广泛，因此本文在建模时只写了模型中的重要部分。最后将精度较好的模型进行概率等权重融合得到精度更高的多分类模型。

5. 1 模型的建立

5.1.1 传统机器学习文本分类方法

朴素贝叶斯文本分类：

运用贝叶斯分类方法[2], 对留言的一级标签进行分类。先确定留言文本中关键词出现在特征词类中的概率, 然后利用贝叶斯概率公式求解后验概率, 根据概率的大小得出分类结论。朴素贝叶斯分类模型的表示方法如下：

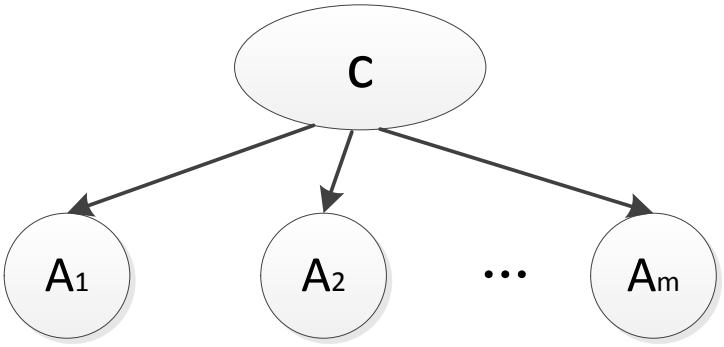


图 11：朴素贝叶斯分类模型

其中，C 是类别节点，A1，A2，…，Am 表示类别节点 C 下文本表示的 m 个属性结点。使用朴素贝叶斯分类器对文本进行分类的过程如下：

(1).首先使用特征向量来表示文本类别，这样构造每个文本类别的特征向量空间就成为首要工作，我们就可以把训练集分成 m 类特征向量空间，每个文本类别拥有了一个唯一的表示该类别文本的特征向量空间。

(2).本文用 $P(s_i \mid x)$ 来表示待分类文本 x 属于类别 s_i 的概率，那么文本分类的关键就是求出使 $P(s_i \mid x)$ 取最大值的类别。

(3).根据 $P(A_m \mid C) = \dfrac{P(A_m)P(C \mid A_m)}{\sum P(A_i)P(C \mid A_i)}$，用 $P(s_i \mid x)$ (i=1,2,…,m)计算每个类别的条件概率。

(4).文档所属类别就是条件概率最大的类别。可以用公式表示为：$p(s_k \mid x) = \max\{p(s_i \mid x), ..., p(s_m \mid x)\}$，则 $x \in s_k$。

5.1.2 深度学习文本分类方法

1. Text-CNN 模型

文本分类模型中，从最经典的 Text-CNN 模型[3]开始，深度学习模型在文本的分类任务上就具有广泛的应用。Text-CNN 模型的网络结构如下图所示：

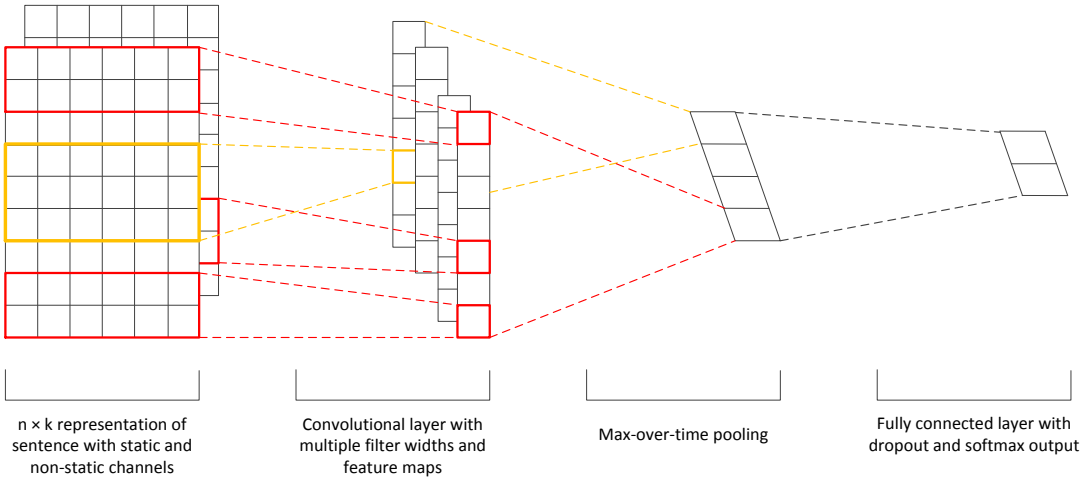


图 12：Text-CNN 网络结构图

整个模型由输入层、卷积层、池化层、全连接层这个四部分构成。

(1).输入层：

Text-CNN 模型的输入层中输入的是一个定长度的文本序列，根据分析语料库样本的长度来指定一个输入的序列长度 N，比 N 短的样本序列则需要填充，比 N 长的序列则需要截取。最终输入层输入的是文本序列中各个词汇对应的词向量。

(2).卷积层：

在 Text-CNN 模型中的卷积核一般是有多个不同尺寸的组成。卷积核的高度，称作窗口值，可以理解为 N-gram 模型中的 N，就是利用的局部词序的长度，窗口值是一个超参数，需要在任务中进行尝试，一般选取为 2~8 之间的值。

(3).池化层：

Text-CNN 模型的池化层中使用的是最大值池化（Max-pool），一方面使得模型的参数减少了，另一方面又保证了在不定长的卷积层的输出上获取一个定长的全连接层的输入。

(4).全连接层：

全连接层的作用就相当于是分类器，Text-CNN 模型使用了只有一层隐藏层的全连接网络，相当于把卷积与池化层提取的特征输入到一个 LR 分类器中进行分类。

2. Text-RNN 模型

在通过数据处理后得到的词向量作为词向量的输入，对于每一个输入序列，可以在 RNN 的每一个时间步长上输入文本中一个字词的向量表示，计算出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字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，由于输入文本的长度为 L，所以要经历 L 个时间步。其网络结构流程如下：

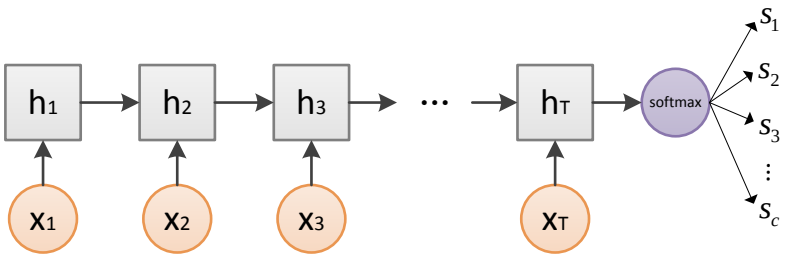


图 13：Text-RNN 网络结构图

图中，X_i表示输入文本，h表示 RNN 神经元，S_i表示输出的分类情况。

在 Text-RNN 模型[4]中，一般取前向或者反向 LSTM 在最后一个时间步长上隐藏状态，之后进行拼接，在经过一个 softmax 层(输出层使用 softmax 激活函数)进行一个多分类；或者取前向或反向 LSTM 在每一个时间步长上的隐藏状态，对每一个时间步长上的两个隐藏状态进行拼接，然后对所有时间步长上拼接后的隐藏状态取均值，再经过一个 softmax 层(输出层使用 softmax 激活函数)进行一个多分类(2 分类的话使用 sigmoid 激活函数)。

3. FastText 模型

FastText 的模型架构[5]与 word2vec 的 CBOW 非常相似，都是基于分层 softmax 函数，都是包含输入层、隐藏层和输出层三层架构。FastText 的网络如下：

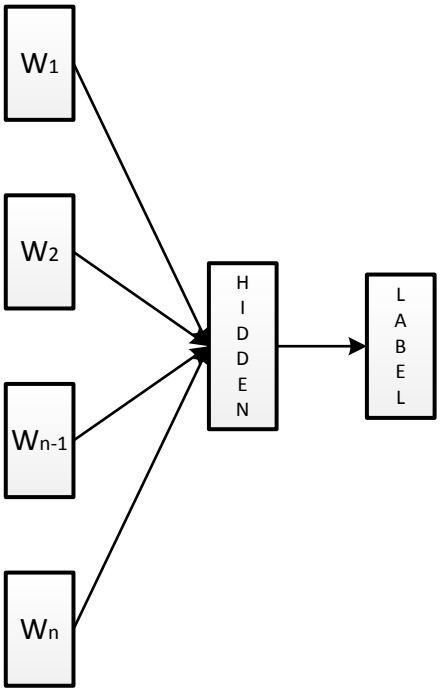


图 14：FastText 模型网络

(1).模型的架构

FastText 的模型则是将整个留言文本作为特征去预测文本的类别。将输入层中的词和词组构成特征向量，再将特征向量通过线性变换映射到隐藏层，隐藏层求出解最大似然函数值，然后根据每个类别的权重和模型参数构建 Huffman 树，将 Huffman 树作为输出。

(2).层次 Softmax

在某些文本分类任务中类别很多，计算线性分类器的复杂度高。为了改善运

行时间，FastText 模型使用了层次 Softmax 技巧。层次 Softmax 技巧建立在 Huffman 编码的基础上，对文本标签进行编码，能够提高模型的效率。

(3).N-gram 特征

FastText 模型对输入的文本词序列加入了 N-gram 处理，用来处理部分词顺序丢失的问题。其做法是把 N-gram 作为一个词，用 embedding 向量表示，在进行计算模型隐层时，把 N-gram 的 embedding 向量也加进去求和取平均。最后通过上面这三个方面完成对 FastText 的建模。

4. Bert 模型

Bert 模型[6]的重要部分是根据双向 Transformer 编码器实现的，其模型结构如图下所示：

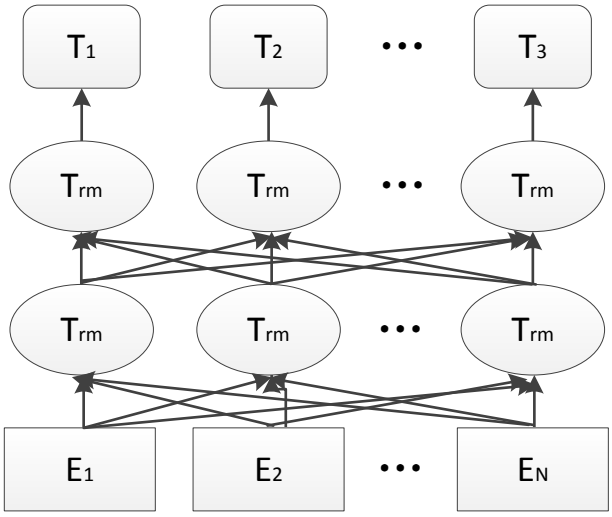


图 15：BERT 模型结构

图中的 $E_1, E_2, ..., E_N$ 表示字词的文本输入，经过双向的 Transformer 编码器，就可以得到文本的向量化表示，即文本的向量化表示主要是通过 Transformer 编码器而实现的。模型的整体结构分为：Embedding、Transformer Encoder、Loss 优化三个部分。

(1). Embedding：Bert 输入的数据是两文本（句话），通过“sep”分隔符标记，分割符前称为前段 E_a，分隔符后面的称为后段 E_b，每次都是输入两句（文本），目的是为了后面计算对 classification 的 loss。另外，对 position embedding 的引入是为了处理 attention 机制中忽略词的顺序。

(2). Transformer Encoder：首先扩展输入的 embedding 维度，一个维度作为 Key，一个维度作为 query，一个维度作为 Value。然后进行 multi_head 划分，对于扩展后的每一个维度，都需要进行划分，图中例子本文进行的是两个头的划分。

(3). Loss 优化：在 word embedding 的时候说到，是用“sep”标记分割的两句话作为模型的输入，因此这两句话有两种可能是，一、两句话来源于同一篇文章，属于上下文的关系，classfication 为 True；二、如果这两句话不相关，不属于上下文的关系，则 classfication 为 False。因此整个模型就是通过优化这两个任务的损失，来训练整个模型。

5.1.3 多模型融合

在上面的模型中计算出各自的精度 F1 值后，本文将精度较高的模型进行融合处理，以提高模型的 F1 值，从而根据最高的 F1 值确定本文最终的多分类模型。

5. 2 评价方法

问题一研究的问题属于分类问题，分类问题最常用的评价指标包括精确率 P、召回率 R 以及 F1 值，它们的计算公式如下面三条。

(1).查准率 P 是指分类器预测为正且预测正确的样本占所有预测为正的样本的比例，计算公式如下：

$$P = \frac{TP}{TP + FP}$$

(2).查全率 R 是指分类器预测为正且预测正确的样本占所有真实为正的样本的比例，计算公式如下：

$$R = \frac{TP}{TP + FN}$$

其中，TP：正例预测正确的个数 FP：负例预测错误的个数，TN：负例预测正确的个数，FN：正例预测错误的个数。

(3). F 值是综合了 P 和 R 的一个指标，一般计算公式如下

$$F_1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2P_iR_i}{P_i + R_i}$$

5. 3 实验参数及结果分析

5.3.1 实验参数

附件二中总共数据 9210 条数据，本文将数据进行随机打乱，经过数据预处理之后，对于较短的文本用空格补齐，较长文本截取前 128 个字，并从中选取 8000 条数据作为训练集，1000 条作为测试集，剩余 210 条作为测试集。系统训练时相关参数设置如表 4 所示：

表一：系统训练时相关参数

参数	参数值
学习速率	0.00005
学习速率减缓因子	0.99
最大梯度范数	3
批次大小	32
最大文本长度	128
训练集大小	8000
验证集大小	1000
测试集大小	210

5.3.2 结果分析

根据模型的建立部分，本文将从传统机器学习和深度学习，单模型和多模型融合两个角度进行对比分析，确定最终的高精度的分类模型。

1.单模型

从传统的深度学习模型与深度学习中的四个模型进行 F1 值比较，各个模型的线下分数如下：

表二：线下模型得分表

模型	precision	recall	F1
朴素贝叶斯	0.849605	0.829165	0.835048
FastText	0.881973	0.901333	0.897143
TextCNN	0.899243	0.912128	0.905219
TextRNN	0.898721	0.911543	0.906433
Bert	0.909243	0.922128	0.915219

由上表可知：采用传统的朴素贝叶斯分类器得到的分类结果明显差于基于深度学习的训练模型，其中采用 Google 的 Bert 模型得到的效果最好，其 F1 值为 0.915。而 FastText，TextCNN 和 TextRNN 模型的分类效果差不多，本文继续进行多模型的融合观察是否更高精度更适合本题的分类模型。

2.多模型融合

在原有模型的基础上，为了探究进一步优化模型精度的可能性，本文采用控制变量的方法，将多个模型训练得到的结果进行等概率权重融合，进一步提升了模型的精度，具体结果如下表：

表三：模型融合得分表

模型 1_F1	模型 2_F1	融合后分数	变量
TextCNN_0.8992	FastText_0.8971	0.9052	模型差异
TextCNN_0.8992	TextRNN_0.9064	0.9087	模型差异
TextCNN_0.8992	TextCNN_0.9021_数据增强	0.9107	数据差异

由模型融合结果，我们得到如下两点结论：

1.通过第一行和第二行的对比中我们发现：模型差异越大提升越多。TextCNN 和 FastText 都是采用的 CNN 结构，只不过 FastText 结构更加简单。虽然 FastText 和 TextRNN 融合后都能提升模型的精度。但是 TextRNN 的提升效果更加显著。

2.从第一行和第三行的对比之中我们可以看出，数据的差异越大，融合的提升越多，采用数据增强的方式，有助于提升数据的差异性，对融合后模型的精度提高也很大。

3.融合结果

模型融合的方法有很多，最常见的比如 bagging，stacking 等。在这次比赛中，由于计算资源不足，导致模型训练计算时间非常长，所以对于我们来说，比较难做 stacking，所以最后使用了多模型线性加权的方法进行模型的融合。具体为每个模型都输出一个 shape=[n_test_sample, n_label]的预测概率矩阵，然后把每个模型的概率矩阵加权平均，对于每个样本，取概率最大的值作为最后的预测结果。最终模型主要是 4 个模型的等概率加权融合，**F1 值在 0.9243**。

因此，最终本文选择将 TextRNN、TextCNN、FastText、Bert 进行等概率加权模型作为第一题的多分类模型。

六．群众问题挖掘

由问题的二的分析，本文首先使用 TF-IDF 进行留言的特征选择，然后使用 DBSCAN 算法对留言进行分类处理，然后依据改进的 Reddit 算法对留言时间和留言点赞情况进行热度评价的量化，最后使用 TextRank 算法对已通过分类排名好的留言进行留言主题的挑选，从而获得留言明细。

6.1 特征选择

本文使用 TF-IDF[7]算法做特征选择。TF-IDF 是一种用来信息搜查的常用加权方法。

TF_{ij} 表示第 i 个关键词在第 j 个文档中出现的频率。其表示为，词频 TF_{ij} =某个词在文章的出现次数 n_{ij} /文章的总词数 $\sum_k n_{ik}$ 。即：

$$TF_{ij} = \frac{n_{ij}}{\sum_k n_{ik}}$$

DF 表示在每条留言中，出现某个关键词的留言个数。IDF 是对一个字词普遍重要性的度量。某一个特定词语的 IDF，是由总文件数目|D|除以包含该词语之文件的数目($1+DF_j$)，再将所得到的商取对数，即：

$$IDF_j = \log\left(\frac{|D|}{1+DF_j}\right)$$

如果一个字词越常见，则其分母就越大，逆文档频率就越小可能就越接近于0。分母加1的目的是为了排除分母为0的情况（表明所有文档都没有包含该词）。

TF-IDF 计算公式如下：$TF-IDF_{ij} = TF_{ij} * IDF_j$

TF-IDF 是与一个字词在留言中的出现次数成正比，与该词在整个留言文本语言中的出现次数成反比。因此，自动提取关键词的算法就非常清楚了，就是计算出文档中每个字词的 TF-IDF 值，然后按照降序排列，取排在最前面的几个词。

6.2 热点问题分类

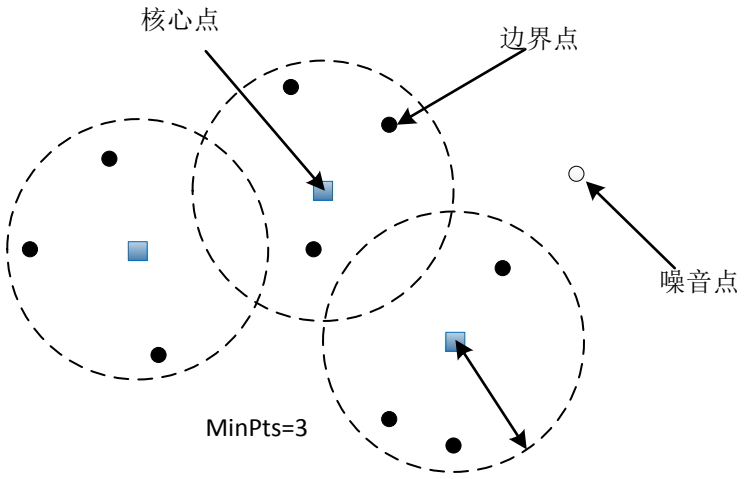
DBSCAN[8]是一种基于密度的空间聚类算法。该算法是将具有足够密度的区域划分为簇，并在具有噪声的空间数据库中发现任意形状的簇，它将簇定义为密度相连的点的最大集合。在本文中的作用也就是将留言问题分成几类（簇）。

1.DBSCAN 算法中的数据点可分为三类：

①核心点：将样本 X_i 的领域内至少包含了 MinPts 个样本，则称为样本点 Xi 为核心点

②边界点：若样本 X_i 的领域内包含的样本数目小于 MinPts 个样本，但是它在其他核心点的领域内，则称为样本点为边界点核心点

③噪音点：既不是核心点也不是边界点的点。



此处这里有两个参数，其中一个参数是半径 eps，另一个是指定的数目 MinPts，就是聚类的类别数。

2.算法步骤

步骤一：首选随机选取一个点，然后找到离这个点距离小于等于 eps 的所有的点。如果距离起始点的距离在 eps 之内的数据点个数小于 min_samples，那么这个点被标记为噪声。如果距离在 eps 之内的数据点个数大于 min_samples，则这个点被标记为核心样本，并被分配一个新的簇标签。

步骤二：然后访问该点的所有邻近点（在距离 eps 以内）。如果它们还没有被分配到一个簇，那么就将刚刚创建的新的簇标签分配给它们。如果它们是核心样本，那么就依次访问其领近点，以此类推。随着簇的逐渐增大，直到在簇里的 eps 距离内没有更多的核心样本为止。

步骤三：选取另一个还没有被访问过的点，并重复相同的过程。

3.参数的设置

①如果 eps 设置得非常小，则意味着没有点是核心样本，可能会导致所有点被标记为噪声。

②eps 设置得非常大，可能会导致所有的点形成单个簇。

③虽然不需要显示设置簇的个数，但设置 eps 可以隐式地控制找到 eps 的个数。

④使用 StandarScaler 或 MinMaxScaler 对数据进行缩放，有时更容易找到 eps 的较好取值。因为使用缩放技术将确保所有特征具有相似的范围。

6. 3 热度评价指标的确定

6.3.1 热度评价指标

在前面的过程中，本文已经使用 DBSCAN 算法将群众留言按照特定地点或特定人群进行分类，但是作为政务服务是有向后处理优先级的，因此需要对这些分类后的群众留言进行热度评价。本文对热度的评价是基于时间和点赞数来进行定义的，因为热度事件的发生往往具有时间效应和普遍性，因此本文使用改进的 Reddit[9]的算法进行热度评价。

1.群众留言的新旧程度

群众的留言一般反映现阶段热点问题，可能不止一个人进行留言反馈，因此

根据分类的结果，在每一篇留言中必然有最先提出该留言的群众Y_{k0}，故可以表示留言的新旧程度为：t=留言时间−该类留言最早时间，即：

$$t = Y_{ki} - Y_{k0}$$

t的单位为秒，用 unix 时间戳计算。有上面公式可知，一旦留言发出，t就是固定值，不会随时间改变，而且留言越新，t值越大

2.赞成票与反对票的差

群众的留言往往具有普遍性，可能不只一人遇到，常常会得到其他人的共鸣，因此对于每一条留言获得其他人的投票情况为，赞成票与反对票的差g=赞成票−反对票

3.投票方向

这里借用数学里面的符号函数的知识，α是一个符号变量，表示对留言的总体看法。如果某一留言的赞成票较多，α就取+1；如果某一留言的反对票较多，α就取−1；如果该留言的赞成票与反对票相等，α就取 0，即

$$\alpha = \begin{cases} 1, & g > 0 \\ 0, & g = 0 \\ -1, & g < 0 \end{cases}$$

4.留言的受肯定程度

$$\beta = \begin{cases} |g|, & g \neq 0 \\ 1, & g = 0 \end{cases}$$

β表示留言的点赞数和反对数之间差的绝对值。如果某一留言的点赞数等于反对数，那么β就等于 1。

综合以上的几个变量，最终确定的热度评价得分指标为：

$$score = \lg\beta + \frac{\alpha t}{4.5*10^5}$$

6.3.2 热度评价指标合理性解释

1.第一个部分$\lg\beta$，这个部分表示，留言的点赞数超过反对数的数量越多，得分越高。

此处，这里使用的是以 10 为底的对数，意味着$\beta=10$可以得到 1 分，$\beta=100$可以得到 2 分。也就是说，前 10 个投票人与后 90 个投票人（甚至再后面 900 个投票人）的权重是一样的，即如果一个留言反映的问题特别受到大众的认同，那么越到后面获得的点赞数，对得分越不会产生影响。当留言的反对数超过等于点赞数，$\beta=1$，因此这个部分等于 0，也就是不产生得分。

2.第二个部分$\frac{\alpha t}{4.5*10^5}$，这个部分表示，$t$越大，得分越高，即新的留言的得分会高于之前的留言。它起到自动将旧留言的排名往下拉的作用。由于政务留言处理时效性相较社交平台要求较低，因此本文在这里降低了对时间因素的权重

α的作用是用来产生正分和负分。当赞成票超过反对票时，得分为正；当赞成票少于反对票时，得分为负；当两者相等时，这一部分的得分为 0。这就保证了得到大量净赞成票的留言，会排在前列；得到大量净反对票的留言，会排在最后。

6.4 热度留言信息主题选择

TextRank 算法[10]是由谷歌的网页重要性排序算法 PageRank 算法改进而来的， PageRank 算法根据网页之间的链接关系构造网络，而 TextRank 算法根据词之间的共现关系构造网络；PageRank 算法构造的网络中的边是有向无权边，而 TextRank 算法构造的网络中的边是无向有权边。TextRank 算法的核心公式如下，其中用于表示e_{ji}两个节点(字词)之间的边连接具有不同的重要程度：

$$WE(V_i)=(1-f)+f\times\sum_{V_j\in In(V_i)}\frac{e_{ji}}{\sum_{V_k\in out(V_j)}e_{jk}}WE(V_j)$$

其中，f 表示阻尼系数，即按照超链接进行浏览的概率，一般取经验值为 0.85。$(1-f)$ 表示浏览者随机跳转到一个新网页的概率。

使用 TextRank 算法提取关键词和关键词组的具体步骤如下：

(1).对已给定文本(序列)按照整句或词组进行分割，即$T=[E_1,E_2,...,E_m]$；

(2).对于每个句子或词组$E_i\in T$，经过数据预处理得到的词向量，构建词图 G=(V,E)，其中 V 为节点集合，由以上步骤生成的词组成，然后采用共线关系构造任意两个节点之间的边：两个节点之间存在边仅当它们对应的词在长度为 K 的窗口中共现，K 表示窗口大小，即最多共现 K 个单词，一般 K 取 2~5；

(3).依据上面的公式，迭代计算各个节点的权重，直至收敛；

(4).对节点权重进行倒序排序，从中得到最重要的 t 个字词词组,作为 top-t 的关键词；

(5).根据得到的 top-t 关键词，在原始文本中进行标记，若它们之间形成了相邻的词组，则作为关键词组提取出来。

从给定文本中提取关键句时，将文本中的每个句子分别看作一个节点，如果两个句子有相似性，则认为这两个句子对应的节点之间存在一条无向有权边，衡量句子之间相似性的公式如下：

$$S(e_i,e_j)=\frac{|w_k|w_k\in e_i\bigcap w_k\in e_j|}{\log(|e_i|)+\log(|e_i|)}$$

其中，e_i、e_j表示两个句子(短语)，w_k表示句子中的词。

根据以上相似度计算公式循环计算任意两个节点之间的相似度，设置阈值去掉两个节点之间相似度较低的边连接，构建出节点连接图，然后迭代计算每个节点的 TextRank 值，排序后选出 TextRank 值最高的几个节点对应的句子作为关键句（字词）。

6.5 实验参数及结果分析

6.5.1 实验参数

1. 本文通过对数据进行筛选和处理，剔除日期格式不正确和一些异常数据，总共有 4327 条数据，经过筛选后剩余 4111 条。

2. 本文采用的自定义消歧词典如下：

表四：自定义消歧词典

原词	替换词
中海	中海国际
凉塘	凉塘路
限购	限售
泉塘街	泉塘街道
国际广场	国际商业广场
未完工	延期
小区幼儿园	幼儿园

3. 采用的用户字典表如下：

表五：用户字典表

县郡小区	泉塘街道	A 市	楚龙街道
普惠性	洋湖中学	经开区	碧桂园
活动板房	泉塘中学	月湖市场	非法诈骗

4. 停用词表采用的是哈工大中文停词表，其中因为留言中数字区分了不同的地区信息，因此停词表中去除了数字。

6.5.2 结果分析

1.热点问题分类情况

本文使用 DBSCAN 算法进行留言分类，其中参数设定半径为 0.45，最少包含点数 5 个，最后得到 8 类留言热点簇，其中聚类效果图如下：

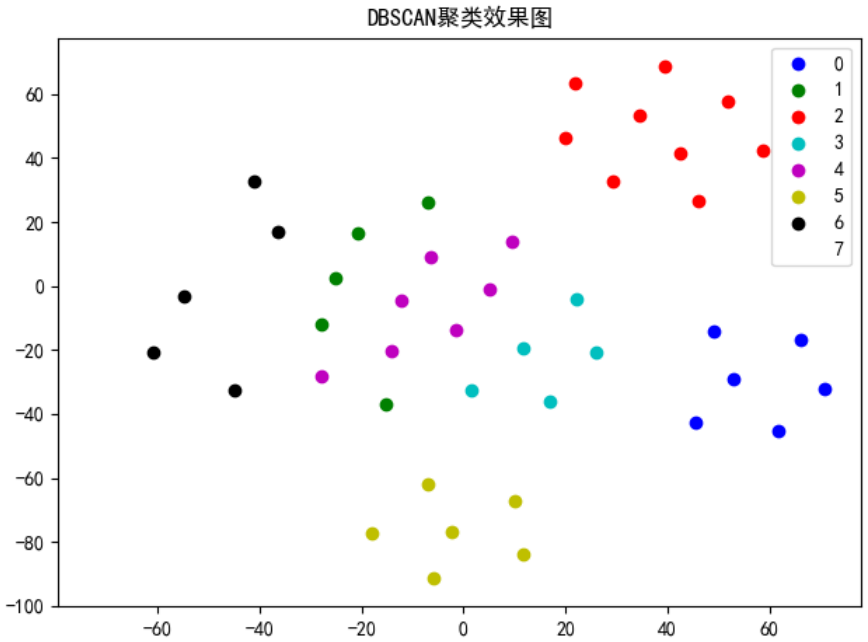


图 16：DBSCAN 聚类效果图

根据 DBSCAN 的聚类效果，本文将所有的留言分类 8 类热点问题，然后从每一类留言中提取出相应的特征词汇，并计算这些特征词汇对应的比例，绘制出了对应的饼状体如下：

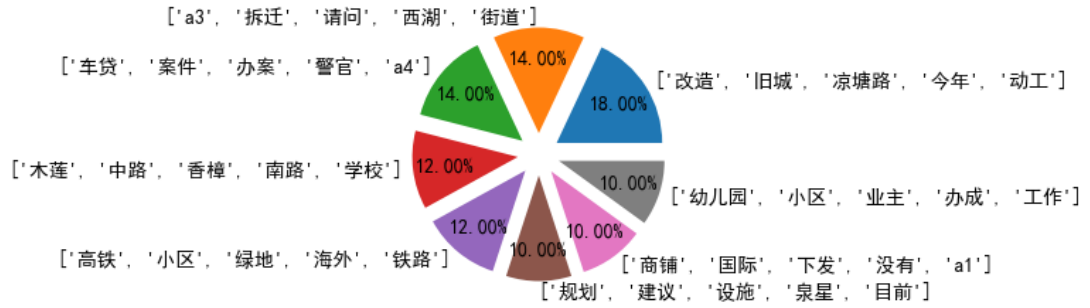


图 17：聚类饼状图

从上图可以看出，特征词汇中反映了热点留言中的相关地点，时间，人物信息，对热点留言新闻整体具有一定的代表性，能够反映一段时间内群众集中反映的问题。其中大概可以看出“旧城改造”，“街道拆迁”和“车贷办案”这三个方面占比较大，可能是由于这三类问题比较严重。

2.热点问题排行

最后本文根据热度排行算法，综合考虑了留言的时效性，点赞、反对情况等信息计算出每类留言的平均热度值，并使用 TextRank 算法从每一类留言当中选出一条最具代表性的留言标题作为热点的问题的问题描述，最终结果如下表：

表六：热点问题表

热度排名	问题编号	热度值	时间范围	地点/人群	问题描述
1	1	298.5977	2019/01/14 20:23:57 至 2019/07/08 17:16:57	A 市 A4 区	西地省 A 市 58 车贷恶性退出 A4 区立案已近半年毫无进展
2	2	294.9329	2019/02/14 10:07:59 至 2019/09/09 08:20:47	A7 县星沙四区凉塘路	A7 县星沙四区凉塘路旧城改造要拖到何年何月才能动工
3	3	287.9818	2019/02/21 12:02:17 至 2019/09/12 08:30:47	A3 区西湖街道茶场村	A3 区西湖街道茶场村五组何时启动拆迁
4	4	279.0084	2019/08/23 14:21:38 至 2019/09/06 18:36:16	A4 区绿地海外滩小区	A4 区绿地海外滩小区距长赣高铁最近只有 30 米不到合理吗
5	5	274.95	2019/08/09 16:47:36 至 2019/08/26 13:00:06	A 市经开区泉星公园	给 A 市经开区泉星公园项目规划进一步优化的建议

由上表可以看出：热度排名前 5 的热点问题中“旧城改造”，“街道拆迁”和“车贷办案”是排名在前 3 的，这与上面的特征词汇占比饼图中占比较高的前三类是一致的。同时热点问题的热点值也可以清晰低在表中看见。

3.留言热点详情

将所有的热点问题相关的留言绘制成词云如下：

图 18：热点词云图

由词云图可以看出留言反映的问题主要体现在“小区”和“业主”等方面。然后根据排名前五的热点问题将所有的留言进行处理分析，然后在 4111 条留言问题中找到对应前 5 的热点问题详情，下面只展示出部分热点留言详情，其中具体的热点问题留言详细表见附件。

表七：热点问题留言详细

热度排名	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	220711	A0003168 2	请书记关注 A 市 A4 区 58 车贷案	2019/2/21 18:45	尊敬的胡书记：您好!A4 区 p2p 公司 58 车贷，非法经营近四年。在受害人要求下，于去年 8.20 立案侦察，至今已 6 个月整。未发一字立案公告和案件进展财产处置通报，全国仅此一家……	0	8 2 1
…	…	…	…	…	…	…	…
2	250512	A0003562 6	A7 县星沙凉塘路旧城改造究竟要拖到何年何月才能开始？	2019/9/2 14:32:27	星沙的旧城改造今年就要全部结束了，而凉塘路的几排房子改造却在两年前被要求停止改造后就无人问津，个中原因众说纷纭……	0	6
…	…	…	…	…	…	…	…
3	274242	A0005160 8	反映 A3 区西湖街道茶场村拆迁问题	2019/1/21 22:52:19	第一，请问胡书记，人民路过江隧道何时修？第二，请问，我们 A3 区西湖街道茶场村五组的村民何去何从？……	0	0

七．答复意见的评价

从所要解决的问题和问题分析出发，本文首先根据留言详情和答复意见来量化答复的相关性，完整性，可解释性和时效性，具体的量化过程如下。然后使用BP 神经网络进行评价，最后使用线性加权得到每天答复的综合评价值。

7. 1 模型的建立

7.1.1 评价指标的确定

1.答复文本的相关性

首先对附件四中的群众留言详情和答复意见进行数据预处理得到词向量，然后使用 TF-IDF 的留言详情中关键字及其权重向量 $Q_i=(x_{i1},x_{i1},...,x_{in})$，其中 $x_{i1},x_{i1},...,x_{in}$ 表示第 i 个留言详情文本 D_i 中各个关键词在 D_i 中所占的权重。同样使用 TF-IDF 的答复意见中关键字及其权重向量 $A_j=(x_{j1},x_{j1},...,x_{jn})$，其中 $x_{j1},x_{j1},...,x_{jn}$ 表示第 j 个留言文本 A_j 中各个关键词在 A_j 中所占的权重。这里 TF-IDF 算法在问题二中进行详细的叙述，此处不再赘述。

然后文本的相关性根据余弦相关性进行计算得到，即：

$$\cos(Q,A)=\frac{\sum_{k=1}^{n}X(k,Q)\bullet X(k,A)}{\sqrt{\sum_{k=1}^{n}X(k,Q)^2}\bullet\sqrt{\sum_{k=1}^{n}X(k,A)^2}}$$

其中，$X(k,Q)$ 表示第 k 个留言详情中关键字及其权重向量 Q，$X(k,A)$ 表示第 k 个答复意见中关键字及其权重向量 A。由该公式，其余弦值越大，则表明文本的相关性就越高。

2.答复文本的完整性

本文对答复意见的完整性的理解是，政务回答者提供的答复意见能够完整的包含群众留言详情中所提出的问题。因此，本文使用答复意见包含对留言详情问题解答的程度来量化文本的完整性。

具体做法为：分别提取留言详情和答复内容中的关键词组，并将每个留言中的关键词和答复内容的关键词进行对比得到一个相似度序列，取最大的相似度作为该关键字的得分，最后计算该留言关键词组的平均得分，作为第 i 个答复文本完整性的度量 ρ_i。其计算公式为：

$$\rho_i=\frac{1}{n}\max_{1\le i\le n}\theta_{ij}$$

其中，θ_{ij} 表示的每条留言详情中的第 i 个关键字与该条留言答复中的第 j 个关键字的相关性，实际上 θ_{ij} 表示的是一个相关系数矩阵，其行数为留言详情中的关键字的个数数值，列数为答复意见中的关键字的个数数值，θ_{ij} 的计算公式就是上面的 $\cos(Q,A)$。具体答复完整性的计算方法如下图：

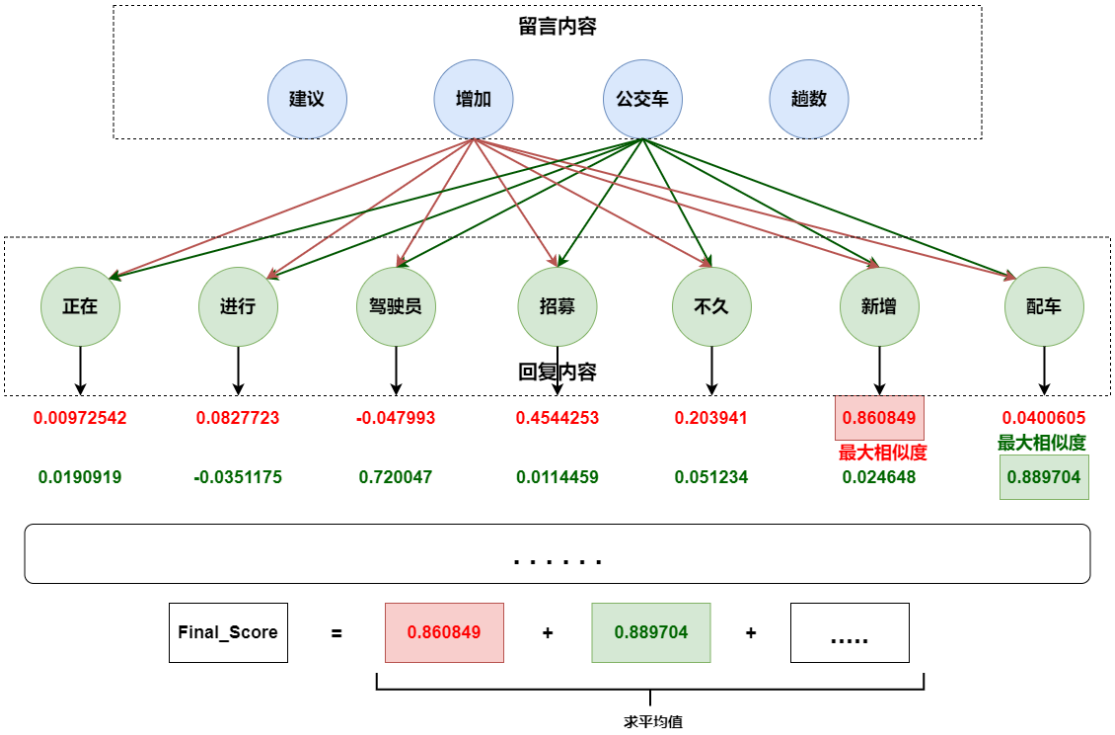


图 19：答复完整性计算方法

3.答复文本的可解释性

答复意见文本越长，则回答就可能越丰富越完整，同时越可能提供事实材料进行解释。如果答复者给出的是推测事实反而会给人一种不可靠不充分的感觉。所以要体现出答复文本的可解释性，本文从留言详情文本和答复文本中推测字词两个角度进行量化。即有：

$$\lambda=0.5*\frac{\text{答复文本词向量长度}}{\text{留言文本词向量长度}}-0.5*\frac{\text{答复文本推测性字词数量}}{\text{答复文本所有字词数量}}$$

本文在这里给予文本长度和推测词的权重均为 0.5，答复文本越长越能体现文本的可解释性，而答复文本中出现“可能，应该，也许”等推测性字词反而会降低文本的可解释性。即计算公式如下：

$$\lambda=0.5*\frac{l_{ik}}{l_{ij}}-0.5*\frac{\sum_{p=1}^{n}n_{ip}}{\sum_{k=1}^{n}n_{ik}}$$

其中，l_{ik} 表示第 i 条答复意见的词向量长度，l_{ij} 表示第 i 条留言详情的词向量长度，n_{ip} 表示第 i 条答复意见的第 p 个推测性词汇，$\sum_{k=1}^{n}n_{ik}$ 表示第 i 条答复意见中所有字词数。此处，n_{ip} 属于 0-1 变量。

$$n_{ip}=\begin{cases}1, & \text{当第p个字词是推测性字词时}\\ 0, & \text{当第p个字词不是推测性词时}\end{cases}$$

4.答复文本的时效性

政务回答者针对群众留言的回答是具有时间间隔，一般都是过了几个工作日才开始回复，因此答复意见与留言详情之间的时间间隔往往就体现出工作的效率，

本文将其定义为时效性。答复时间间隔 t 为答复意见b_{ik}与留言时间b_{ij}之差，即

$$t = b_{ik} - b_{ij}$$

根据答复时间的间隔，本文定义答复的效率如下：

$$\eta = \begin{cases} 0.8, & t \in (0,7] \\ 0.6, & t \in [8,15] \\ 0.4, & t \in [16,23] \\ 0.2, & t \in [24,31] \end{cases}$$

由于留言时间和答复时间都是精确到秒，因此需要先将时间转化为天，然后进行计算，在转化为天的过程会存在一些四舍五入的过程。

7.1.2 BP 神经网络

BP 神经网络[11]是神经网络中使用最广泛和最具代表性的一类模型。从结构上来说，BP 网络是多层网络模型，分为输出层、隐含层、输入层，各层之间实行全连接。BP 网络模型实现了多层学习的构想，加入给定 BP 网络一个输入模式，它由输入层神经元传到隐含层神经元，然后经过隐含层神经元逐层处理后再发到输出层神经元，由输出层神经元处理后产生一个输出模式，这时完成了一个逐层状态更新过程，被称为向前传播。其三层结构如下：

第一层（输入层）：将输入引入神经网络

$$Out_i^{(1)} = In_i^{(1)} = x, i = 1,2\cdots n$$

第二层（隐藏层）：

$$\begin{cases} In_j^{(2)} = \sum_{i=1}^{n} w_{ij}^{(1)} \times Out_i^{(1)} & j = 1,2\cdots l \\ Out_j^{(2)} = f(In_j^{(2)}) \end{cases}$$

其中 $f(x)$ 为传递函数，这里采用的是正切 *sigmoid* 函数： $f(x) = \tanh(x)$

第三层（输出层）：

$$y_{(k)} = Out_k^{(3)} = In_k^{(3)} = \sum_{j=1}^{i} w_j^{(2)} \times Out_j^{(2)}, k = 1,2\cdots n$$

这里 m,n,l 分别代表输入层节点数，输出层节点数，隐层神经元个数。

根据本问题可以直接确定 $m=7, n=1$。至于 l 不能直接确定，这里采用经验公式： $l = \sqrt{m+n} + a$，其中， n 为输入层神经元个数， m 为输出层神经元个数， a 为[1,10]之间的常数。 根据上式可以计算出神经元个数为 4-13 个之间来确定其初始值。

下图为根据问题三建立的 BP 神经网络评价模型结构图：

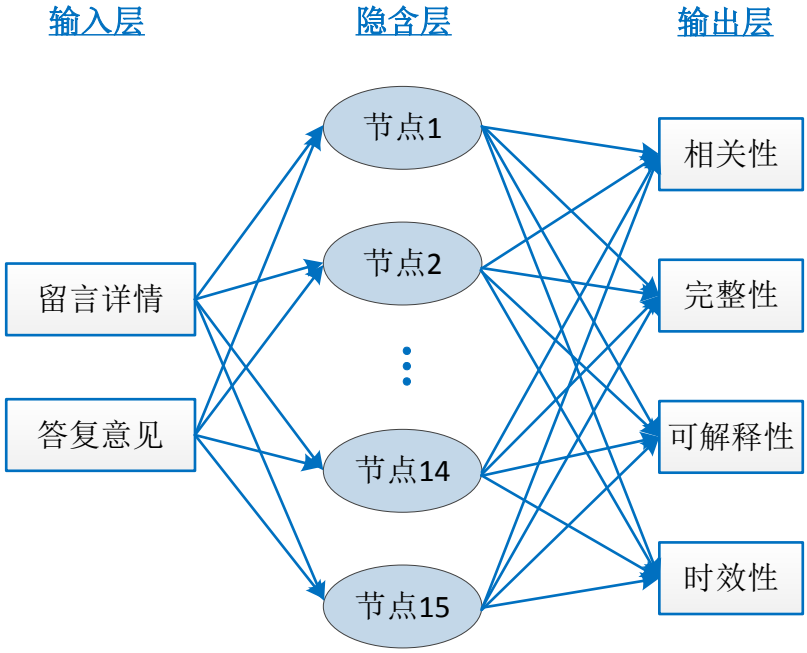


图 20：BP 神经网络结构图

其中，选用留言详情处理后的词向量和作为输入层，评判标准作为输出层，此神经网络的输入层神经元为 2 个，输出层神经元为 4 个。确定了隐含层神经元数量是 15，为了使误差更小，速度更快，在设置初始参数时，选择了迭代次数 epochs 为 10000 次，学习率 lr 为 0.05，目标值 goal 为 10^{-100}。

其后，用 python 求解 BP 神经网络评价模型，然后对这四个输出进行线性加权融合得到一个综合指标，即：

$$F = \sum_{j=1}^{4} \xi_j w_j$$

其中，ξ_j 表示这四个评价指标的权重，w_j 表示这个四个变量的输出值，本文在这里是使用等概率线性加权。

7. 2 结果分析

（1）根据附件 4 中的 2816 条数据按照训练集：测试集=3 ：1 的比例放入神经网络中进行训练，将输出的四维数据：时效性（timeliness），可解释性（interpretability），相关性（correlation），完整性（completeness）线性加权作为每条回复的得分指标（score），最后我们选出了得分最高的十条留言，其综合评价结果如下：

表八：模型结果表

rank	留言编号	timeliness	interpretability	correlation	completeness	score
1	97307	0.9164	0.9817	0.9911	0.9650	0.9636
2	9128	0.9178	0.9671	0.9933	0.9478	0.9565
3	6448	0.9534	0.8873	0.9530	0.9276	0.9303
4	50409	0.9191	0.8459	0.9787	0.9535	0.9243
5	24691	0.9571	0.9061	0.8952	0.8741	0.9081
6	25528	0.8794	0.8939	0.8138	0.8965	0.8709
7	117669	0.7365	0.9431	0.9467	0.8233	0.8624

8	36363	0.8497	0.7594	0.8910	0.9373	0.8593
9	68414	0.9676	0.8163	0.8676	0.7784	0.8575
10	105651	0.9043	0.8895	0.9032	0.7284	0.8563

其中这排名前 10 的答复的具体留言详情见附件。

（2）根据求解结果，对所有留言的得分进行统计如下图：

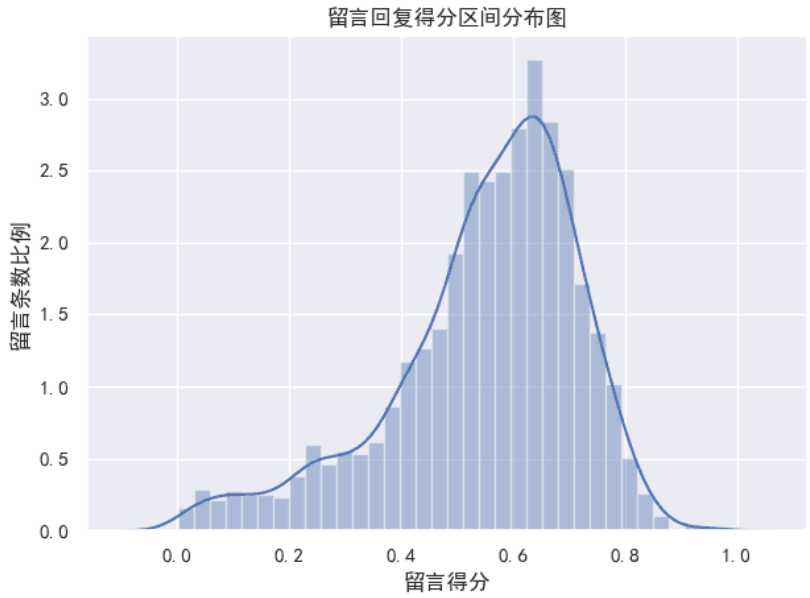


图 21：留言回复得分区间分布图

由上图可知，可以清晰的看到附件 4 中所有的答复意见的得分分布，大部分的答复意见的综合得分是处在 0.5~0.8 之间，一方面说明该政务工作者的工作服务情况是符合群众需要的，也具有一定的提升空间。同时，从所有答复得分的趋势线来看，这些得分情况是服从正态分布的，这也反映出模型的合理性。

八．模型的评价

8.1 模型的优点

1.在问题一中使用深度学习中的多模型融合，避免数据量不够而引起模型的过拟合状况，同时融合后的模型获得更高的 F-Score，使得分类精度更高，更适合问题的多分类情况。

2. 问题二中使用改进的 Reddit 算法，原因是政务留言处理时效性相较社交平台要求较低，改进后降低了对时间因素的权重，更加符合实际。

3.使用 DBSCAN 算法，其聚类的速度快且能够有效地处理噪声点和发现任意形状的空间聚类，不需要人工胡划分聚类个数，聚类出的各个簇没有偏倚。

8.2 模型的缺点

1.问题一中只是使用了附件 2 中已知一级标签进行分类，就是通过已知的关系进行模型的训练求解。没有考虑到三级标签，所以从这个角度来看，问题一中使用的模型还可以更深入。

2.本文对问题三中答复意见的相关性、完整性、可解释性和时效性的定义虽然是合理的，但是对答复的可解释性和时效性的量化过程中存在着一定的主观性，为避免这种情况，可以扩大数据集的维度或者结合问卷调查进行量化处理。

九．模型的改进与推广

9.1 模型的改进

1.对文本问句语义提取

可以将附件 1 中的三级标签理解为一个三元组组成的集合实体，实体关系和实体。留言详情就相当于实体，需要分析留言详情与各级标签的关系。

因此可以首先基于语义分析的方法通过将实体和联系进行识别和标注将自然语言形式的问句转换为 lambda 表达式或依存组合语义树等逻辑表达形式。通过对这种逻辑表达形式，进行向量化表示，从而进一步加强，对语义的聚焦。

2.数据集的扩充

附件 4 提供的数据集，主要是用来评价答复的质量，但实际上要想综合出评价答复的质量，需要从群众特征、回答者特征和社会情感等多方面进行综合评价，这样就可以使用层次分析法进行贴合实际的评价。所需的这三个方面的数据可以使用 python 爬取获得，然后进行建模分析。由于某些原因，本文没有实现这一改进。

9.2 模型的推广

随着互联网时代的飞速发展，网络上有海量的文本信息，想要处理这些非结构化的数据就需要利用 NLP 技术，NLP 在信息提取、文本情感分析和个性化推荐等多方面的应用非常广泛。

十．参考文献

[1]唐明,朱磊,邹显春.基于 Word2Vec 的一种文档向量表示[J].计算机科学,2016,43(06):214-217+269.

[2]姜天宇,王苏,徐伟.基于朴素贝叶斯的中文文本分类[J].电脑知识与技术,2019,15(23):253-254+263.

[3]明建华,胡创,周建政,姚金良.针对直播弹幕的 TextCNN 过滤模型[J/OL].计算机工程与应用:1-8[2020-05-07].http://kns.cnki.net/kcms/detail/11.2127.TP.20200110.1720.012.html.

[4]伍逸凡,朱龙娇,石俊萍.人工神经网络在信息过滤中的应用[J].吉首大学学报(自然科学版),2019,40(03):17-22.

[5]代令令,蒋侃.基于 fastText 的中文文本分类[J].计算机与现代化,2018(05):35-40+85.

[6]段丹丹,唐加山,温勇,袁克海.基于 BERT 的中文短文本分类算法的研究[J/OL]. 计 算 机 工 程 :1-12[2020-05-07].https://doi.org/10.19678/j.issn.1000-3428.0056222.

[7]张波,黄晓芳.基于 TF-IDF 的卷积神经网络新闻文本分类优化[J].西南科技大学学报,2020,35(01):64-69.

[8]郭艳婕,杨明,侯宇超,孟铭.基于相似性度量的改进 DBSCAN 算法[J].数学的实践与认识,2020,50(06):164-170.

[9]阮一峰.基于用户投票的排名方法[EB/OL]. http://www. cajcd.edu.cn/pub/wml.txt/980810-2.html,2012-03-29.

[10]尤苡名.基于 TextRank 的产品评论关键词抽取方法研究[J/OL].软件导刊 :1-5[2020-05-07].http://kns.cnki.net/kcms/detail/42.1671.TP.20191126.1610.076.html.

[11]白宝光,范清秀,朱洪磊.基于 BP 神经网络的高新区公共服务质量评价模型研究[J].数学的实践与认识,2020,50(03):154-163.