“智慧政务”中的文本挖掘应用

摘 要

随着各种网络问政平台的出现，政府机构收集到的各类社情民意相关的文本数据量急速攀升。为了更好地划分留言以及整理热点问题，本文采用了自然语言处理和文本挖掘的方法，来对网络问政平台中出现的各类群众问政留言记录进行自动化分类，挖掘其中的热点问题，以及评价政府对群众留言意见的答复。

对于各类群众问政留言的自动化分类问题，本文使用了基于词频-逆文档频率 (TF-IDF) 的方法来对群众问政留言记录的主题与内容进行向量化表达，实现文本空间到向量空间的坐标转化。在此基础上，本文建立了基于使用高斯核的支持向量机 (SVM) 和朴素贝叶斯思想的分类模型来对问政留言进行分类并在此之后采用了 F-Score 对这两种分类模型进行评价，结果高达 0.92。爬取政务网站数据对原数据集做了扩展，完善了分类种类。在新数据集上验证了模型的泛化能力。

对于热点问题挖掘问题，本文首先采用了词频-逆文档频率(TF-IDF) 的方法将留言的文本内容转换为 N 维空间中的向量，然后采用基于密度的 DBSCAN 算法将相似的留言问题聚集在一起。在此之后考虑留言问题的点赞数与反对数，并采用基于威尔逊思想的得分算法计算热点问题集中各类问题的热度指标，选取热度指标前 5 名作为热点问题填入题目要求的“热点问题表”。

对于评价政府部门对留言问题的回复，本文从答复的相关性、完整性、可解释性、时效性角度出发，提出了匹配度、通顺度、及时度三个指标。其中匹配度通过计算词向量的最高相似度之和得到，通顺度通过依存句法分析得到的语法通顺度和语义通顺度组合而成，及时度通过分析提问时间与答复时间之间的工作日天数得到。最后采用动态分配系数的方法将三个指标组合在一起得到了用于评价答复质量的总分。

关键词：支持向量机；朴素贝叶斯；DBSCAN；威尔逊得分；依存句法分析

Abstract

With the emergence of various online questioning platforms, the amount of text data related to various social conditions and public opinions collected by government agencies has rapidly increased. In order to divide the messages and sort out the hot issues better, this article uses natural language processing and text mining methods to automatically classify various types of mass questioning message records appearing on the network questioning platform, and to dig out the hot issues, Evaluate the government's response to the comments of the masses.

For the automatic classification of various types of people's messages, this article uses a method based on TF-IDF to vectorize the content of the masses' political message records to achieve text space to vector space Coordinate transformation. On this basis, this paper establishes a classification model based on the SVM with Gaussian kernel and Naive Bayesian idea to classify the questionnaire and then uses F-Score to classify these two classification models Evaluation, the result is as high as 0.92. Crawling government website data expands the original data set and improves the classification. The generalization ability of the model is verified on the new data set.

For hotspot problem mining, this paper first uses TF-IDF method to convert the text content of the message into a vector in N-dimensional space, and then uses the density-based DBSCAN algorithm to gather similar message problems in together. Then considering the number of likes and number of the message, utilize the scoring algorithm based on Wilson Score to calculate the heat index of various question.

Regarding the evaluation of government departments' responses to message questions, this paper puts forward three indicators: matching, syntactic, timely degree in terms of relevance, completeness, interpretability and timeliness. The matching degree is obtained by calculating the sum of the highest similarity of the word vector, the syntactic degree is combined by the grammatical and the semantic syntactic degree through the dependency syntax analysis, and the timely degree is obtained by analyzing the number of working days between the question time and the answer time. Finally, the dynamic allocation coefficient method is used to combine the three indicators to obtain a total score for evaluating the quality of the response.

Keyword: support vector machine; naive bayes; dbscan; willson score; dependency parsing analysis

目 录

1. 挖掘目标

文本分类是自然语言处理(NLP)的一个基本任务，传统的基于概率的机器学习算法适合对文本进行处理，而且也已经有了不错的应用成果：垃圾邮件的分类、情感分析、AI 法官等。

本文的目标是根据各网络群众问政平台的群众留言问题，结合已有的传统机器学习算法以及一些中文文本处理工具，完成以下三个目标：

⑴ 对群众留言进行分类，以方便政府精准地下派问题至相关部门机构。

⑵ 提取留言信息中的热点问题，并构造合适的算法计算热点问题的热度指标，政府部门可以根据热度指标有针对性地、高效地处理热点问题。

⑶ 为政府对留言问题的回复进行评价，从答复的相关性、完整性、可解释性等角度对 答复意见的质量给出评价。

2. 群众留言分类

2.1 任务流程

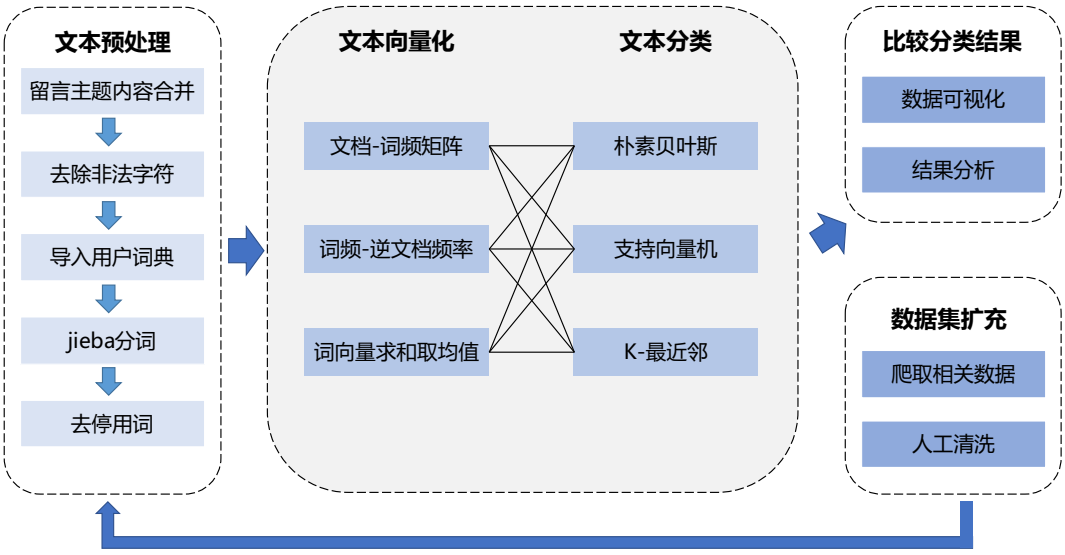

图 1 任务一流程

本用例主要包括如下步骤：

步骤一：数据预处理。由于题目给出的数据是来自互联网，经过爬虫软件爬取下拉的，里面的内容比较杂乱。可以利用 Jieba 中文分词软件进行中文分词，然后去除分词列表中的非法字符、空白字符、非中文字符、停用词等操作。在分词的过程中，导入本文所准备的用户词典以提升分词的精确度。

步骤二：文本向量化。在步骤一结果的基础上，利用 TFIDF 或 Word2Vec 等技术对分词进行向量化。

步骤三：构建分类模型。将总数据集按照二八原则，将 80%的数据分为训练集来训练分类模型，剩下 20%的数据作为测试集。

步骤四：结果分析与数据集扩充。测试分析不同向量化方法与分类模型的运行结果，并引入爬取的数据完善原数据集分类，测试扩展数据集的分类结果。

2.2 数据预处理

2.2.1 中文分词

在对留言信息进行数据挖掘之前，需要将非结构化的文本信息转化为计算机可以识别的信息。在这一步骤中，输入是题目所提供的附件 1.xlsx 以及附件 2.xlsx，该步骤的任务是从 xlsx 文件中读取所需要的文本信息，如果留言主题、留言详情等内容，然后利用 Python 提供的 Jieba 中文分词工具将每一条留言的语句分割成一个个的词。在 Python 的 Jieba 分词工具中，采用了基于前缀字典实现的高效词图扫描，可以找出基于词频的最大切分组合，而且该分词工具采用了基于汉字成词能力的 HMM 模型，可以很好地发现新词与未登录词[1]。为了更好地提取留言信息中的特殊名词如人群、地点等信息，本文爬取了全国主要的小区名称并且使用了搜狗的细胞词库作为用户自定义词典，以提高 Jieba 分词的准确度。

2.2.2 分词清理

题目给予的留言信息集是来源于互联网络，通过爬虫软件从 HTML 文档中提取而来，中间可能夹杂着如 HTML 标签、制表符、换行符甚至编码以外的未知字符等，我们需要将这些无效的字词从留言中剔除，避免其对结果产生干扰。

去除这些字词一个比较好的方法是使用正则表达式。Python 中提供的 re 库可以很好的完成这个任务。表 1 展示了本文所使用到的一些正则表达式以及对应的作用。

表 1 正则表达式

正则表达式	作用
\?+	清理所有未知字符
\u3000	清理全角空白符
[\t\r\f]	清理空格、制表符、回车、换页符
\n+	清理多个换行符
\s+	清理多个 Unicode 中的空白符

清理结束之后的工作是去除语句中的停用词，停用词指的是如“的”，“那么”，“如果”之类的对文档意义没有帮助的词，将这类字词从语句中剔除以后不影响语义的表达。

最终部分分词结果如图 2 所示。分词结果保存于附录的原始数据集分词结果.xlsx 中。

```
0 ['市', '西湖', '建筑', '集团', '占', '道', '施工', '安全隐患', ...
1 ['市', '在水一方', '大厦', '人为', '烂尾', '多年', '安全隐患', ...
2 ['投诉', '市', 'A1', '区苑', '物业', '违规', '收', '停车费', ...
3 ['A1', '区', '蔡锷', '南路', 'A2', '区华庭', '楼顶', '水箱', ...
4 ['A1', '区', 'A2', '区华庭', '自来水', '好大', '一股', '霉味', ...
```

图 2 留言分词（部分）

2.3 文本向量化

在数据挖掘过程中，我们需要把待挖掘的数据转换为计算机可以计算的数量，也就是说我们需要把留言的文本内容转换为一个有意义的数字或者向量。在任务一中，我们尝试了以下三种将文本转换成向量的方法。

2.3.1 文档词频矩阵

词频即一个词在文档中出现的频率。将所有词库中的词当成列，并初始化向量为[0,0,…,0,0]，长度由词库的词的个数决定，如果文本中出现某一词，那么某一词的数就+1，除以文档总词数后得到词频的向量。

$$词频(TF) = \frac{词在文档中出现的次数}{文档总词数} \tag{1}$$

实际上，词频较小的词其重要程度很多时候往往高于词频较大的词，所以仅仅统计词频的方法在实际中有一定的缺陷。

2.3.2 词频-逆文档频率[2]

由于实际应用中，一篇文档中罕见词往往比常见词蕴含更多的信息，因此在词频（TF）的基础上，引入了逆文档频率（IDF）的概念来衡量单词总体重要性，其值等于文档总数除以包含该单词的文档数量的商再取其商的对数。

$$逆文档频率(IDF) = log\left(\frac{语料库的文档总数}{包含该词的文本数 + 1}\right) \tag{2}$$

最后可以计算出每个词的 TF-IDF（词频-逆文档频率）值：

$$TF - IDF = 词频(TF) \times 逆文档频率(IDF) \tag{3}$$

使用 TF-IDF 对留言文本进行向量化，有助于降低群众留言中常见词语的特征，从而提取出关键的类属信息。

2.3.3 词向量

词向量（Word Embedding）也叫词嵌入，是将每个词语映射到一个向量空间中。对于中文词语，本文使用 Tencent AI Lab 训练好的 Word2Vec 模型[3]，此模型包含 800 多万中文词汇的语义信息，具有较高的覆盖率、新鲜度以及准确性。由于 Word2Vec 的但词向量之间有如下关系：

$$\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} = \overrightarrow{queen} \tag{4}$$

本文通过词向量加和再除以总词数来描述一段留言的语义信息：

$$文档向量 = \left(\sum_{i=1}^{文档词数\,n} 词向量_i \right) \Big/ 文档词数\,n \tag{5}$$

2.4 分类模型

2.4.1 朴素贝叶斯[4]

朴素贝叶斯分类器在样本数据属性值之间的独立性假设下以贝叶斯公式为核心的分类模型。主要是通过数据的先验概率，利用贝叶斯公式计算出其后验概率，然后取后验概率最大的作为该数据的标签[5]。

假设给定了训练数据集$\left(\mathrm{x}^{(\mathrm{i})},\ \mathrm{y}^{(\mathrm{i})}\right)_{i=1,2,3,\ldots,m}$，其中 $\mathrm{x}^{(\mathrm{i})} \in R^n$，而且数据集的每一维属性都是独立的随机变量。这样可以得到训练数据的先验概率

$$\mathrm{P}(\mathrm{X}_1 = \mathrm{x}_1, \mathrm{X}_2 = \mathrm{x}_2, \mathrm{X}_3 = \mathrm{x}_3, \ldots, \mathrm{X}_\mathrm{n} = \mathrm{x}_\mathrm{n} | Y = y) \tag{6}$$

$$= \prod_{\mathrm{j}=1}^{\mathrm{n}} \mathrm{P}\left(\mathrm{X}_\mathrm{j} = \mathrm{x}_\mathrm{j} \middle| \mathrm{X}_1 = \mathrm{x}_1,\ \mathrm{X}_2 = \mathrm{x}_2, \mathrm{X}_{\mathrm{j}-1} = \mathrm{x}_{\mathrm{j}-1}, \mathrm{Y} = \mathrm{y}\right)$$

$$= \prod_{\mathrm{j}=1}^{\mathrm{n}} \mathrm{P}\left(\mathrm{X}_\mathrm{j} = \mathrm{x}_\mathrm{j} \middle| \mathrm{Y} = \mathrm{y}\right) \#(6)$$

根据朴素贝叶斯模型中的关键假设以及概率乘积公式我们可以得到

$$P(Y = y, X_1 = x_1, \ldots, X_n = x_n) \tag{7}$$

$$= P(X_1 = x_1, \dots, X_n = x_n | Y = y) P(Y = y)$$

$$= P(Y = y) \prod_{j=1}^{n} P\left(X_j = x_j \middle| Y = y\right)$$

$$= p_Y(y) \prod_{j=1}^{n} p_{X_j|Y}(x_j|y)$$

由此我们可以计算出

$$\mathrm{P(Y = y, X_1 = x_1, \dots, X_n = x_n) = p(y) \prod_{j=1}^{n} p_j\left(x_j \middle| y\right)} \tag{8}$$

并且得到朴素贝叶斯模型的对数-似然函数

$$\mathrm{l\left(\Omega\right) = \log \prod_{i=1}^{m} p\left(x^{(i)}, y^{(i)}\right)} \tag{9}$$

$$= \sum_{i=1}^{m} \mathrm{logp}\left(x^{(i)}, y^{(i)}\right)$$

$$= \sum_{i=1}^{m} \log\left(p\left(y^{(i)}\right) \prod_{j=1}^{n} p_j\left(x_j^{(i)} \middle| y^{(i)}\right)\right)$$

$$= \sum_{i=1}^{m} \mathrm{logp}\left(y^{(i)}\right) + \sum_{i=1}^{m} \sum_{j=1}^{n} \mathrm{logp_j}\left(x_j^{(i)} \middle| y^{(i)}\right)$$

在朴素贝叶斯模型中我们有两组参数，第一组是 Y 的概率质量函数$p(y) = P(Y = y)$，第二组是指定 Y=y 的前提下随机变量X_j的条件概率质量函数$p_j(x|y) = P\left(X_j = x \middle| Y = y\right)$。那么朴素贝叶斯的核心问题就是将这两组参数集求解出来。因此朴素贝叶斯的数学模型可以归纳为

$$\mathrm{Max} \sum_{i=1}^{m} logp\left(y^{(i)}\right) + \sum_{i=1}^{m} \sum_{j=1}^{n} logp_j\left(x_j^{(i)} \middle| y^{(i)}\right) \tag{10}$$

$$s.\mathrm{t.} \sum_{y} p(y) = 1$$

$$\sum_{x} p_j(x|y) = 1, \forall y, \mathrm{j}$$

$$\mathrm{p}(y) \geq 0, \forall y$$

$$p_j(x|y) \geq 0, \forall j, \mathrm{x}, \mathrm{y}$$

这是一个带等式约束的最优化问题，可以使用拉格朗日乘子法来解决。朴素贝叶斯最终的解为：

$$\mathrm{p}(y) = \frac{count(y)}{m} = \frac{\sum_{i=1}^{m} 1\left(y^{(i)} = y\right)}{m}, \forall y \tag{11}$$

$$p_j(x|y) = \frac{count_j(x|y)}{count(y)} = \frac{\sum_{i=1}^{m} 1\left(y^{(i)} = y \wedge x_j^{(i)} = x\right)}{\sum_{i=1}^{m} 1(y^{(i)} = y)}, \forall x, \mathrm{y}, \mathrm{j}$$

可以看出朴素贝叶斯是以频率作为指标来对模型进行训练的，如果测试集中存在某个训练集中没有出现过的 y，那么可能会出现 $\frac{0}{0}$ 的情况，这显然是不合理的。为了解决这个问题，可以引入拉普拉斯光顺（Laplace Smoothing）。

引入拉普拉斯光顺后的朴素贝叶斯模型的解为

$$p(y) = \frac{count(y)}{m} = \frac{\sum_{i=1}^{m} 1\left(y^{(i)} = y\right) + 1}{m + k}, \forall y \tag{12}$$

$$p_j(x|y) = \frac{count_j(x|y)}{count(y)} = \frac{\sum_{i=1}^{m} 1\left(y^{(i)} = y \wedge x_j^{(i)} = x\right) + 1}{\sum_{i=1}^{m} 1(y^{(i)} = y) + v_j}, \forall x, \mathrm{y}, \mathrm{j}$$

其中 k 是所有可能的 y 的数量，v_j是所有可能的第 j 个特征的数量。引入拉普拉斯光顺后依旧可以保证概率和为 1，求出来的值作为概率是合法的。

2.4.2 高斯核的软边距支持向量机[4]

支持向量机也是一种经典的机器学习分类算法，广泛地用于二分类问题。对于多分类问题可以采用一对一（one to one）或者一对多（one to more）的形式来解决。支持向量机的核心是在 N 维空间中寻找一个超平面将数据点分割开来并且使得每个点到超平面距离的最小值最大。为了解决数据集线性不可分或者过拟合的现象，支持向量机引入了核函数与正则项来解决。核函数可以在低维空间中计算高维空间的点积，正则项可以允许 SVM 误分类的存在。本文具体采用了高

斯函数(RBF)作为核函数的软边距支持向量机来实现对文本的分类。

给定数据集 $\left(\mathrm{x}^{(\mathrm{i})},\ \mathrm{y}^{(\mathrm{i})}\right)_{i=1,2,3,\ldots,m}$，其中 $\mathrm{x}^{(\mathrm{i})} \in R^n, y \in -1,1$。SVM 的任务是找到一个超平面 $\mathrm{y} = \omega^T \mathrm{x} + \mathrm{b}$。这样可以计算出每个点到超平面的距离为

$$\gamma^{(\mathrm{i})} = \frac{y^{(i)}\left(\omega^T x^{(i)} + b\right)}{|\omega|} \tag{13}$$

SVM 的数学模型可以表示为

$$\max_{\gamma,\omega,b} \ \min_i \gamma^{(i)} \tag{14}$$

$$\text{s.t.} \quad y^{(i)}\left(\omega^T x^{(i)} + b\right) \geq \gamma|\omega|, \forall i$$

由于对 ω 和 b 的缩放不影响答案，我们可以缩放 (ω, b) 使其满足

$$\min_{\mathrm{i}} \ \mathrm{y}^{(\mathrm{i})}\left(\omega^{\mathrm{T}} \mathrm{x}^{(\mathrm{i})} + \mathrm{b}\right) = 1 \tag{15}$$

这样可以将 SVM 的数学模型简化为

$$\min_{\omega,b} \ \frac{1}{2}|\omega|^2 \tag{16}$$

$$s.t. \quad y^{(i)}\left(\omega^T x^{(i)} + b\right) \geq 1, \forall i$$

为了避免过拟合的情况，可以向这个模型中引入正则项 $\sum_i \xi_i$，得到软边距 SVM 的数学模型：

$$\min_{\omega,b} \ \frac{1}{2}|\omega|^2 + \mathrm{C}\sum_i \xi_i \tag{17}$$

$$\text{s.t.} \quad y^{(i)}\left(\omega^T x^{(i)} + b\right) \geq 1 - \xi_i, \forall i = 1, \ldots, \mathrm{m}$$

$$\xi_i \geq 0, \forall i = 1, \ldots, \mathrm{m}$$

这个问题是一个典型的二次规划（QP）问题，可以使用现成的求解器进行求解。为了提高求解的速度，该问题也可以使用拉格朗日对偶函数来求解。

$$L(\omega, b, \xi, \alpha, r) = \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i\left[y^{(i)}\left(\omega^T x^{(i)} + b\right) - 1 + \xi_i\right] - \sum_{i=1}^{m} r_i\xi_i \tag{18}$$

根据 KKT 条件可以推得其对偶问题

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} y^{(i)}y^{(j)}\alpha_i\alpha_j < x^{(i)}, x^{(j)} > \tag{19}$$

$$s.t. \quad 0 \le \alpha_i \le C, \forall i = 1, \dots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

最终的解为

$$\omega^* = \sum_{i=1}^{m} \alpha_i^* y^{(i)} x^{(i)} \tag{20}$$

$$b^* = \frac{\sum_{i:0<\alpha_i^*<C}\left(y^{(i)} - \omega^{*T} x^{(i)}\right)}{\sum_{i=1}^{m} 1(0 < \alpha_i^* < C)}$$

从上述的对偶问题可以看出来，SVM 数学模型中的最优化任务中包含了$< x^{(i)}, x^{(j)} >$这样一个点积的形式，我们可以使用满足 Mercer's Condition 的核函数ϕ来替换它

$$K_{i,j} = K\left(x^{(i)}, x^{(j)}\right) = \phi\left(x^{(i)}\right)^T \phi\left(x^{(j)}\right) =< \phi\left(x^{(i)}\right), \phi\left(x^{(j)}\right) > \tag{21}$$

在普通的 SVM 中默认使用的是线性核函数

$$K(x, z) = x^T z \tag{22}$$

在任务一中本文所使用的是高斯核函数

$$K(x, z) = exp\left(-\frac{|x - z|^2}{2\sigma^2}\right) \tag{23}$$

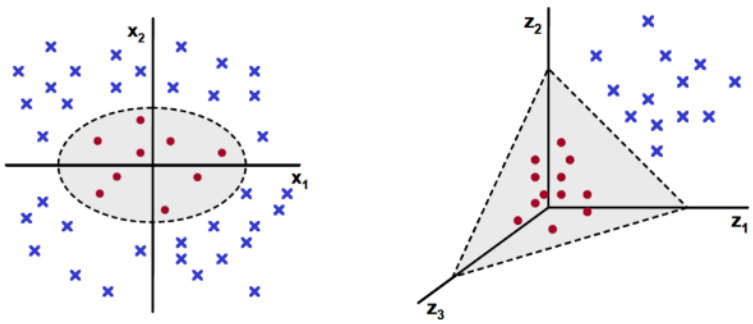

图 3 核函数与高维映射

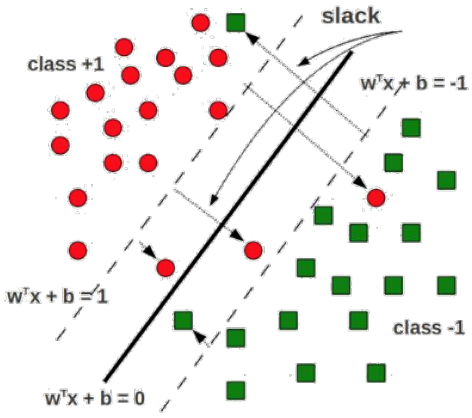

图 4 软边距支持向量机

2.4.3 K-最近邻

K-最近邻（KNN,K-Nearest Neighbors）作为一种有监督分类算法，其算法主体思想就是根据距离相近的邻居类别，来判定自己的所属类别。算法的前提是需要有一个已被标记类别的训练数据集，具体的计算步骤分为一下三步：

1、计算测试对象与训练集中所有对象的距离，本文使用的是欧氏距离；

2、找出上步计算的距离中最近的 K 个对象，作为测试对象的邻居；

3、找出 K 个对象中出现频率最高的对象，其所属的类别就是该测试对象所属的类别。

可见 K-最近邻算法中 K 值的选取对结果影响较大，因此本文通过遍历 K 值来找出效果最好的 KNN 模型。KNN 算法的缺点是，当样本不平衡时，如其中一个类别的样本较大，可能会导致对新样本计算近邻时，大容量样本占大多数，影响分类效果；

2.5 结果与数据集扩充

2.5.1 原始数据集结果

通过对附件 2 数据中 7 个分类的统计，发现数据不均衡性在可以接受的范围之内，因此不需要做过采样或欠抽样来优化训练。

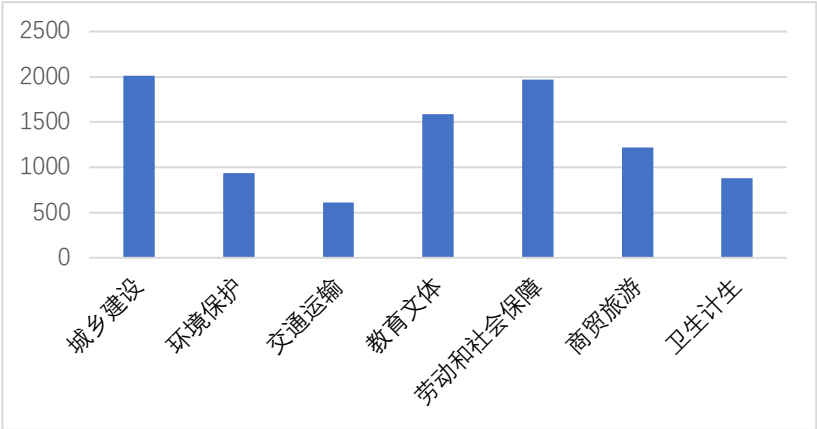

图 5 附件 2 数据分类统计

本文实验了三种向量化方法（文档词频、词频-逆文档频率、词向量），在三种分类模型（朴素贝叶斯、K 最近邻、支持向量机）下的表现，具体结果如图 6 所示。

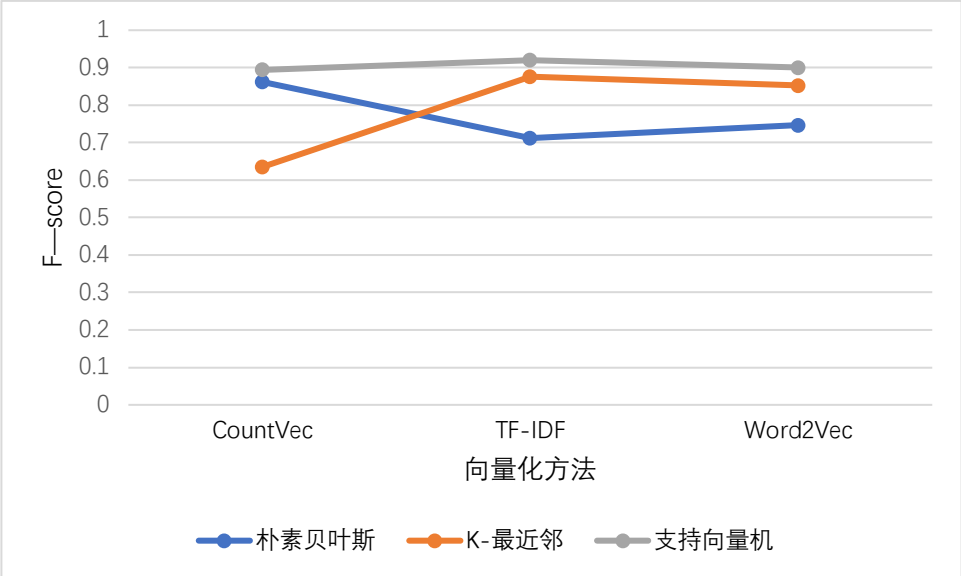

图 6 模型结果

通过结果可以看出，分类模型方面：朴素贝叶斯与 K-最近邻算法的 F-score 值受向量化方法影响较大，而支持向量机一直稳定在高分区域。向量化方法方面：文档词频矩阵效果总体不如 TF-IDF 与词向量优秀。

其中，使用 TF-IDF 向量化方法的支持向量机模型评分最高，平均 F-score 可达 0.92。

表 2 最优模型结果

标签	准确率	召回率	F-score
城乡建设	0.94	0.80	0.86
环境保护	0.94	0.95	0.94
交通运输	0.98	0.91	0.95
教育文体	0.92	0.87	0.89
劳动和社会保障	0.85	0.94	0.89
商贸旅游	0.94	0.93	0.94
卫生计生	0.96	0.95	0.96
平均	0.93	0.91	0.92

2.5.2 扩充数据集结果

实际应用中留言分类远不止上述 7 种，考虑到附件 2 中仅有 7 种分类与现实情况有偏差，我们在原先数据的基础上根据附件 1 的留言分类体系，在政务留言网站相关板块爬取了其他类的留言信息并做了清洗，对数据集进行了扩充和完善。爬取并清洗后的数据保存在附件爬取数据集(清洗后).xlsx 中。

扩充后的数据集规模如图 7 所示：

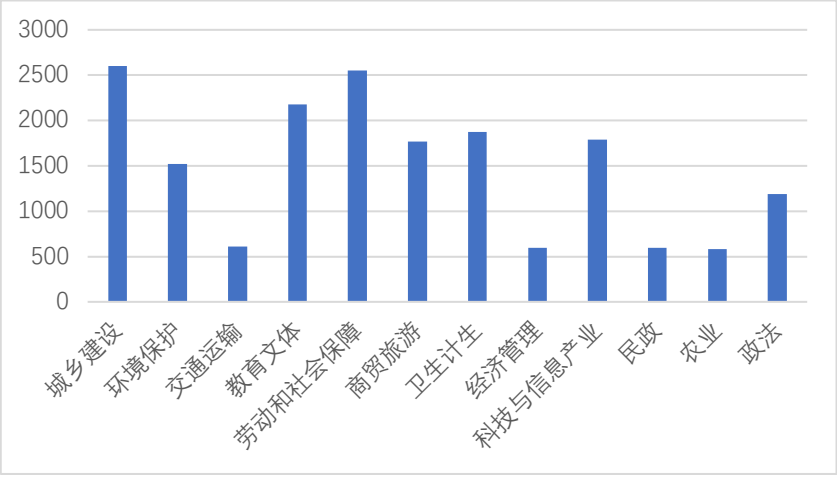

图 7 扩充数据集规模

扩充后的分词结果保存在附件扩充数据集分词结果.xlsx 中。

继续用上述方法将三种向量化方法代入三种分类模型，得到结果如图 8 所示：

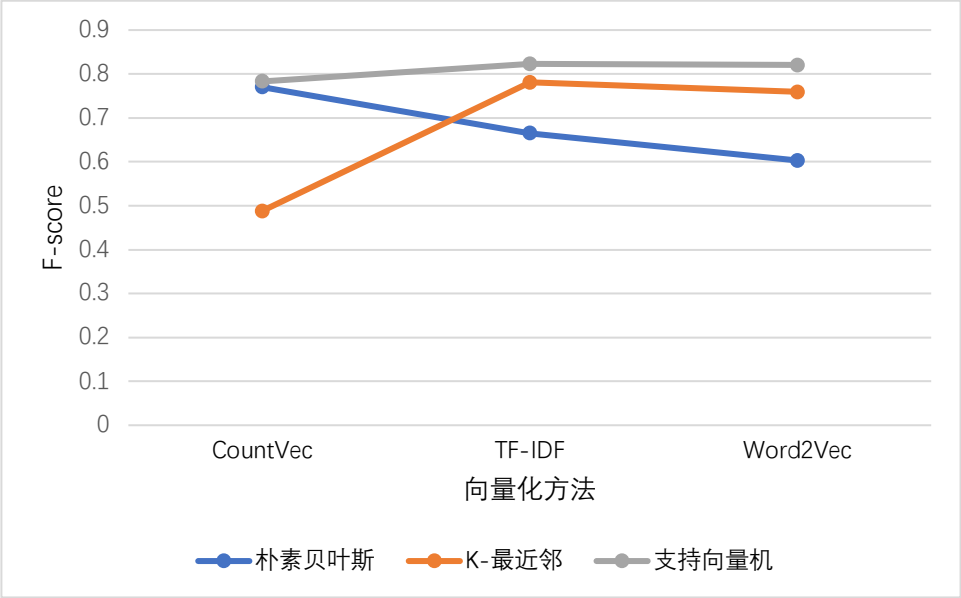

图 8 扩充数据集模型结果

可以看出总体结果与前一实验相似，使用 TF-IDF 向量化方法的支持向量机模型依旧评分最高，F-score 可达 0.83。整体结果较原数据集结果有所降低，推测可能的原因：

①扩充数据集虽然是分模块爬取，但仍存在标签错误的情况，人工清洗时未纠正完全；

②各分类数据规模不均衡，对模型训练产生了影响。

3. 热点问题挖掘

3.1 任务流程

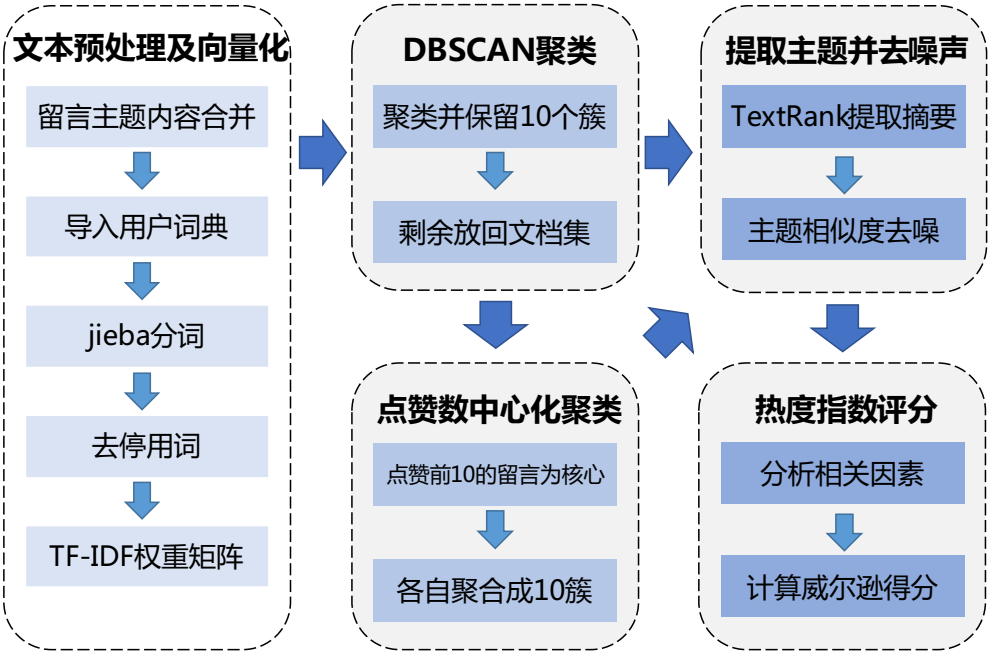

图 9 任务二流程

步骤一：数据预处理，将附件 3 的留言主题与留言详情合并成一段文本并去除标点及空行、加载预先设置好的行政区划用户字典、中文文本分词、停用词过滤，以便后续分析。

步骤二：文本向量化，基于 TFIDF 权重法提取关键词，根据文本的词频以及逆文档频率得出文本向量，构造词汇-文本矩阵。

步骤三：留言聚类，使用 DBSCAN 聚类算法，以文本向量间的余弦相似度为依据，对留言进行聚类，保留留言数量排名前十的簇。再在剩余留言中找出点赞数排名前十的留言，根据文本向量的余弦相似度筛选出其同类留言，并保留相应簇。

步骤四：噪音过滤，将已经成簇的每组留言利用 TextRank 方法求取摘要，计算每篇留言与摘要相似度，迭代过滤留下相似度小于阈值的留言，形成纯净的热点问题组。

步骤五：评分排名取前五。一类留言内的用户数量以及它们的点赞数与反对

数成为热度指标的重要考虑因素。参与留言的用户数量以及点赞数量与热度指标呈正相关，反对数量与热度指标呈负相关。

3.2 数据预处理

3.2.1 添加用户字典

任务要求挖掘出某一时段内群众集中反映的热点问题，由于热点问题的特点是发生在特定地点，作用于特定人群的问题。为了将热点问题聚类并提取出有效问题描述，需要充分利用这两个特点。人群及具体地点（如：经济学院、伊景园小区等）可由 jieba 库的分词及命名实体识别提取。而行政区划（如：A 市、A2 区、K 县等）在分词时会被删除，因此需要将行政区划保存为用户字典，以便更好地分词、提取。由于现实中真实区划具有有限性与固定性，此方法在真实使用上也是行之有效的。

```
0 A7县
1 A7市
2 A7区
3 A7省
4 A7路
5 A7镇
6 A7小区
7 A8县
8 A8市
9 A8区
...
```

图 10 自定义地区字典

3.2.2 中文分词与向量化

在导入用户词典后，对文本进行去标点、分词以及去停用词的处理流程与任务一相同，在此不加赘述。

同样地，需要将分词后的文档转化成 TF-IDF 向量，以方便后续计算。TF-

IDF 的算法与生成向量的具体步骤见 2.3.2 部分。

3.3 留言聚类

这一步骤是希望通过聚类算法将热点问题从留言数据集中抽取出来。对于热点留言，我们考虑两个评价指标：对同一问题的反应数量和留言的点赞数。两者的高低都对问题的热度有直接影响，因此我们采用先对留言采用 DBSCAN 算法聚类选出数目最多的前 10 个问题簇，在对剩余留言中点赞前 10 名的作为聚类核心点，筛选出同类问题。将以上 20 类问题进行统一评价取前五名作为热点问题。

3.3.1 基于密度的 DBSCAN 聚类模型

常见的聚类算法有划分方法、层次方法、基于密度的方法和基于网格的方法。传统基于距离的聚类方法的缺点是只能发现球状簇且在依据向量相似度进行聚类的任务中表现不好。本文选用基于密度的 DBSCAN 算法[6]将文档向量聚类。该方法的合理性见【分析 2.3.1】。

【分析 2.3.1】留言内容虽然数据量庞大，但是存在大量独立或两三条留言描述同一问题的离群点。需要被关注的是小部分具有一定规模并描述同一问题的留言簇，而数据中这样留言簇的数量也无法确定。而基于密度的 DBSCAN 算法可有效过滤处于低密度区域的离群点，聚合任意形状的簇，不必将所有点都聚类，无需指定聚类数目，符合本任务的数据要求，提高了聚类的准确度。

DBSCAN 算法核心思想：

从某个选定的核心点出发，不断向密度可达的区域扩张，从而得到一个包含核心点和边界点的最大化区域，区域中任意两点密度相连。

DBSCAN 算法基本步骤：

输入：样本集$D = (x_1, x_2, \ldots, x_m)$,给定点在邻域内成为核心对象的最小邻域

点数：$MinPts$,邻域半径：ε,样本距离度量方式：余弦相似度.

输出：簇划分C.

1）初始化核心对象集合$\Omega = \phi$，初始化聚类簇数$k = 0$，初始化未访问样本集合$\Gamma = D$，簇划分$C = \phi$

2）对于$j = 1,2,\ldots,m$，按下面的步骤找出所有的核心对象：

a）通过距离度量方式，找到样本x_j的ϵ邻域子样本集$N_\epsilon(x_j)$

b）如果子样本集样本个数满足$\left|N_\varepsilon(x_j)\right| \geq MinPts$， 将样本$x_j$加入核心对象样本集合：$\Omega = \Omega \cup \{x_j\}$

3）如果核心对象集合$\Omega = \phi$，则算法结束，否则转入步骤 4）

4）在核心对象集合Ω中，随机选择一个核心对象o，初始化当前簇核心对象队列$\Omega_{cur} = \{o\}$，初始化类别序号$k = k + 1$，初始化当前簇样本集合$C_k = \{o\}$,更新未访问样本集合$\Gamma = \Gamma - \{o\}$

5）如果当前簇核心对象队列$\Omega_{cur} = \phi$，则当前聚C_s类簇生成完毕，更新簇划分$C = \{C_1, C_2, \ldots, C_k\}$，更新核心对象集合$\Omega = \Omega - C_k$， 转入步骤 3）。否则更新核心对象集合$\Omega = \Omega - C_k$

6）在当前簇核心对象队列Ω_{cur}中取出一个核心对象y'，通过邻域距离阈值ϵ找出所有的ϵ邻域子样本集$N_\epsilon(o')$，令$\Delta = N_\epsilon(o') \cap \Gamma$，更新当前簇样本集合$C_k = C_k \cup \Delta$，更新未访问样本集合$\Gamma = \Gamma - \Delta$，更新$\Omega_{cur} = \Omega_{cur} \cup (\Delta \cap \Omega) - o'$，转入步骤 5）

输出结果为： 簇划分$C = \{C_1, C_2, \ldots, C_k\}$

本文使用余弦相似度计算向量之间的距离，对于最小邻域点数（$MinPts$），通过多次聚类实验我们认为有 8 条反应同一问题的留言就可以算作同一问题类，

因此设置$MinPts = 8$。在设置最小邻域半径（ε）时，根据实验经验得出范围在0.4-0.65 之间，因为超过 0.65 的半径值会使不属于同类留言聚在一起，小于 0.4 则无法识别相同事件的留言。通过遍历参数选择最佳的聚类效果，最终确定$\varepsilon = 0.561$。在保证噪音较小的前提下，最大化聚类的完整性。

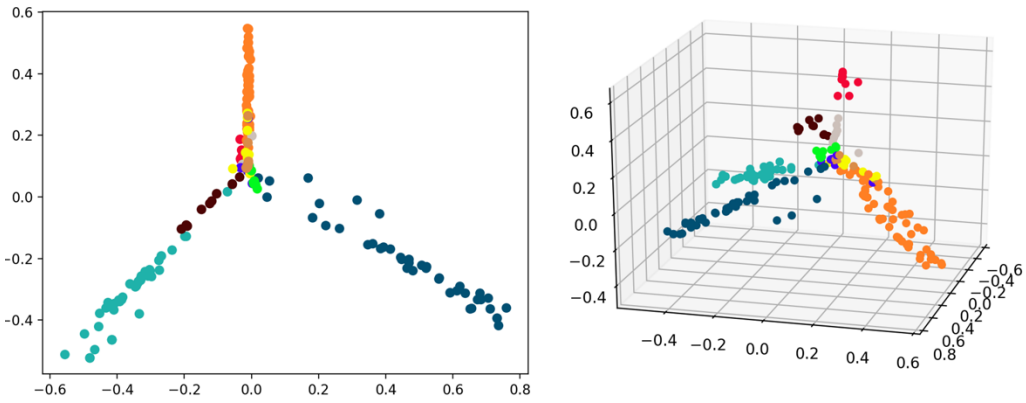

图 11 聚类结果

聚类结果保存在 dbscan 聚类.xlsx 中。

在找出投诉数量最多的 10 个问题后，由于热点问题还有可能出现在点赞数量较高的留言中，所以依然还需要考虑点赞的影响。

3.3.2 点赞数中心化聚类

在对附件 3 的数据进行分析后，我们发现：有些热点问题不能仅仅依靠同类型的留言数量来确定，因为可能有些问题是共有的，那么用户就不会再发布同样的问题，而是对已有的问题点赞来表示关注，因此问题的点赞数量也需要纳入考量。

在点赞数中心化聚类的流程中，我们使用了一次排序，一次聚类，两次清洗的方法得到了最终的聚类结果，具体流程图如图 12 所示。

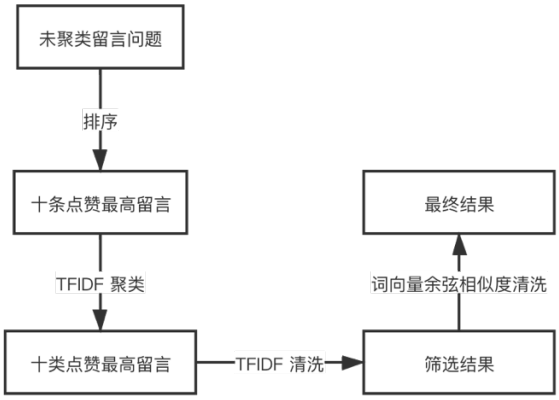

图 12 中心化聚类流程

首先，我们通过排序从上一步骤中未聚类的留言问题中挑选出 10 条点赞数量最高的留言。然后将这 10 条点赞数量最高的留言作为中心，利用 TFIDF 从所有未聚类留言中找出与中心相似的留言，并将它们聚为一类点赞热点问题，共十类。

在之后的过程中，我们一共进行了两次数据的清洗。第一轮清洗以聚类的输出为输入，主要利用 TFIDF 将那些相似度小于 0.1 的留言剔除，把剩余的留言输出。第二轮清洗的时候以第一轮的输出为输入，利用词向量余弦相似度将那些相似度小于 0.42 的留言剔除，并得到清洗完后的聚类结果。对每一类点赞热点均进行上述的两轮清洗，得到最终的聚类结果，保存至点赞中心化.xlsx 中。

需要注意的是，由于 TFIDF 在数据量小的时候性能较差，而在第二轮筛选的时候由于经历了前一轮的筛选，留下来的留言数量并不多。因此我们选择用词向量的余弦相似度来代替 TFIDF 进行相似度计算，以保证筛选的精准度。并且此处的词向量并非分词之后的词向量，由于文本过短，直接分词会导致关键词在词向量中所占维数过少，即使关键词匹配上了，相似度仍然很低。考虑到经过一次数据清洗后，每一类中的文本已经较为相似，并且长度都不长，因此我们以一个字为一个词进行余弦相似度计算，显著提高了最终的筛选精度。

3.4 噪音过滤

上述两个聚类结果都在一定程度上达到的准确完整，但某些簇仍然带有少量噪音，可能会对职能部门解决问题产生干扰，因此仍然需要对聚类结果进行噪音过滤，保障准确性。此时我们可以根据这一类问题的留言提取出他们的关键主题，作为整个问题的摘要内容，然后去和这类问题中的每一条留言内容进行相似度比对，保留那些相似的，去除那些不相似的。我们在去除噪音的时候，将相似度的阈值初值设置的较低，然后经过几轮迭代，不断收紧阈值上限，一轮一轮地剔除那些噪音留言，以避免误删除的情况。

3.5 热度指标

对于一类问题来说，和该问题相关的留言数量、点赞数量以及反对数量都是影响该问题热度指标的因素。但是考虑到存在用户恶意刷留言的情况，我们用参与留言的用户数量替换了留言的数量以避免这种恶意行为对热度指标造成的影响。一类问题的热度指标为归一化后参与留言的用户数量与根据点赞数与反对数计算而来的威尔逊得分之和乘 100。

参与留言的用户数量采用了 Z-Score 归一化方法，经过该方法以后，各个问题参与留言的用户数量符合均值为 0 方差为 1 的标准高斯分布。归一化公式为

$$x_i^* = \frac{x_i - \mu}{\sigma} \tag{24}$$

威尔逊得分[7]是根据点赞反对数量计算评分的一种经典算法。该算法综合考虑了一个问题的评论数以及点赞数，得分越高，质量越高。

$$n = u + v$$

$$p = \frac{u}{n}$$

$$S = \frac{p + \frac{Z_\alpha^2}{2n} - \frac{Z_\alpha^2}{2n}\sqrt{4n(1-p)p + Z_\alpha^2}}{1 + \frac{Z_\alpha^2}{n}} \tag{25}$$

其中 u 表示点赞数，v 表示反对数量。由于在本问题中，可能存在点赞数量和反对数量均为 0 的问题，如果直接使用威尔逊算法计算问题的热度指标会出现分母为零的情况，为了避免，我们假定提出问题的用户会对该问题，这样可以保证该问题的评论数量不为 0，具体的点赞数量设为 k 那么公式中变为

$$n = u + v + k, p = \frac{u+1}{n} \tag{26}$$

3.6 结果

我们得到了五类热点问题，分别为“关于高级技师申报 A 市人才新政购房补贴的相关问题咨询”、“丽发新城小区附近的搅拌站噪音严重扰民”、“投诉 A 市伊景园滨河苑捆绑车位销售”、“A 市五矿万境 K9 县交房后仍存在诸多问题”、“ 西地省 A 市 58 车贷恶性退出，A4 区立案已近半年毫无进展”。 最终聚类效果比较好，热点问题所涉及的地点人群比较统一与突出。结果保存在热点问题表.xlsx 与热点问题留言明细表.xlsx 中。

图 13 是这些热点问题的词云。

图 13 热点问题词云

4. 答复意见的评价

4.1 评价指标说明

在本问题中，我们共选取了三个指标用于评价答复意见的质量，分别是匹配度、通顺度和及时度。

匹配度针对“答复与留言的内容是否相关”、“答复是否完整地解决了留言的困惑”等问题给出了答复意见与留言问题之间匹配程度的数学描述，涵盖了相关性、完整性等角度。

通顺度针对答复意见本身语义、语法的通顺流畅程度给出了对应的数学描述，涵盖了可解释性角度。

留言问题通常具有时效性，即答复意义随时间流逝不断递减。因此我们在任务描述中所涉及的相关性、完整性、可解释性三个角度之外引入了及时性的角度，并针对这一角度提出了第三个指标 —— 及时度，给出答复意见及时性的数学描述。

最后我们将根据三个指标的对应得分动态调整各指标的系数，给出针对答复意见质量的评价总分。

4.2 匹配度

引入匹配度的概念来衡量答复信息的相关性与完整性。留言与答复中心词语语义的匹配度可以衡量留言与答复的相关性，避免答非所问；留言与答复整篇文档词语语义的匹配度可以衡量答复回答留言的完整性，由于留言往往是涵盖多个层次的复杂型问题，因此要确保答复涉及到留言提及的各个方面。

4.2.1 数据预处理

首先将附件 4 中每一条留言的留言主题与留言内容合并，再与相应的答复信

息一起做去标点、分词、去停用词的处理，为保证地点信息的匹配，分词前依然需要加载任务二中用到的用户字典。

对于留言与答复中的每一个分词，使用 Tencent AI Lab 训练好的 Word2Vec 模型加载其对应词向量，使得每篇留言或答复文档转化成 n 个 200 维向量形成的向量序列。方便后续匹配算法计算。

4.2.2 匹配度计算

计算匹配度的算法来自于机器翻译中对齐的思想[8]，不是词与词一一对应，而是词之间近似度的匹配。

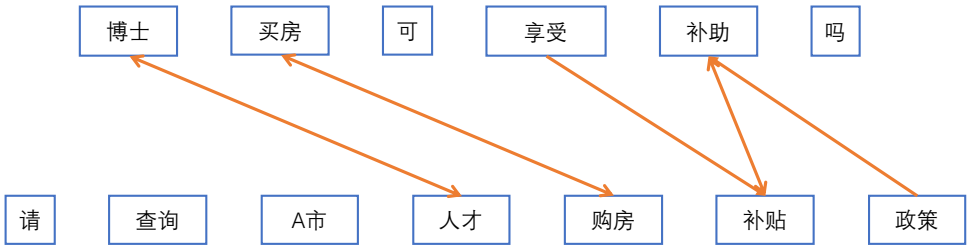

图 14 对齐匹配

具体方法是将留言中每个词在答复中找出与其匹配度最高的词，统计最高的相似度之和，除以留言总词数，作为留言对答复的匹配度。相似度通过词向量的余弦相似度来衡量。将答复中的每个词对留言做同样的相似度匹配，求出答复对留言的匹配度。二者取平均值为最终的文本匹配度具体公式如下：

$$MatchScore = \frac{\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} CosSim_{ij}}{n} + \frac{\sum_{j=1}^{m}\sum_{i=1}^{n} CosSim_{ji}}{m}}{2} \tag{27}$$

最后我们根据公式 27 得到了图 15 所示的数据集中答复匹配度的分布图。

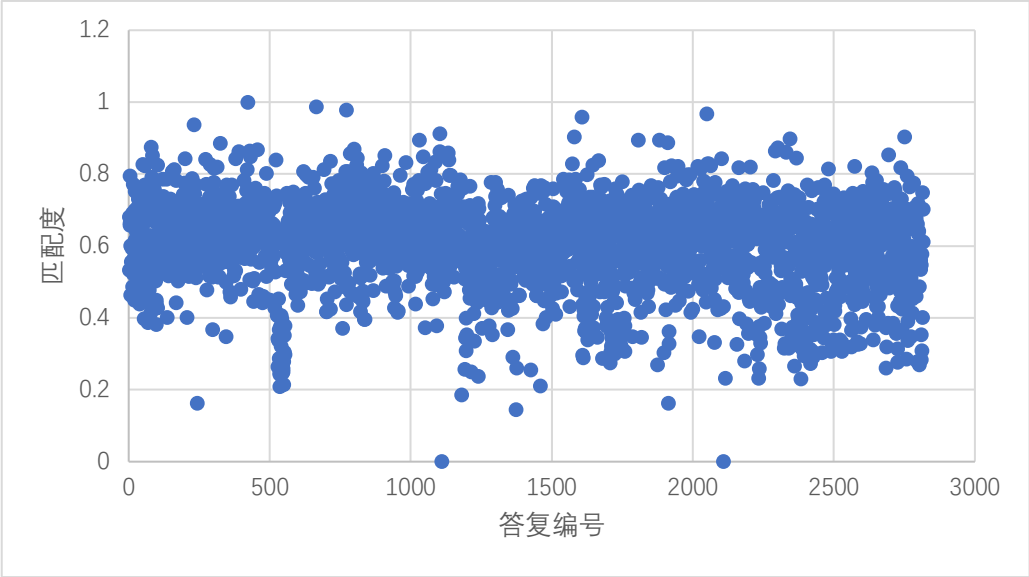

图 15 匹配度分布图

4.3 通顺度

4.3.1 依存句法分析

依存句法分析是一种确定句子的句法结构以及句子中各分词间依存关系的方法。以答复数据中“经调查，C 市高新区双马中心学校 2016 年前没有收到农村贫困寄宿生补助经费。”为例，句法分析结果如图 16 所示。

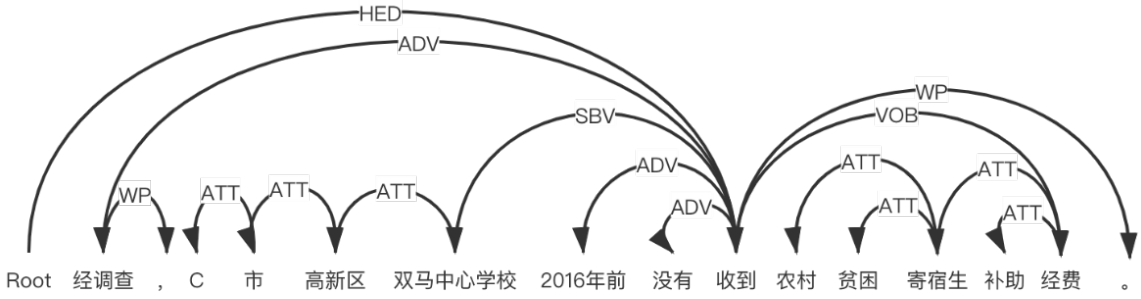

图 16 依存句法分析结果

依存语法分析功能使用了哈工大社会计算与信息检索研究中心研制的语言技术平台（LTP）[9]，其中常见的 15 种依存句法关系如表 3 所示。

表 3: LTP 平台定义的依存关系

关系类型	关系表示	关系类型	关系表示	关系类型	关系表示
主谓关系	SBV	定中关系	ATT	左附加关系	LAD
动宾关系	VOB	状中结构	ADV	右附加关系	RAD
间宾关系	IOB	动补结构	CMP	独立结构	IS
前置宾语	FOB	并列关系	COO	核心关系	HED
兼语	DBL	介宾关系	POB	标点符号	WP

4.3.2 语法通顺度

如果一条答复中的语法错误既多又明显，就会极大地影响提问者对于答复的理解，即这条答复的质量非常低[10]。因此语法通顺度对于答复质量的评价来说就显得非常重要。

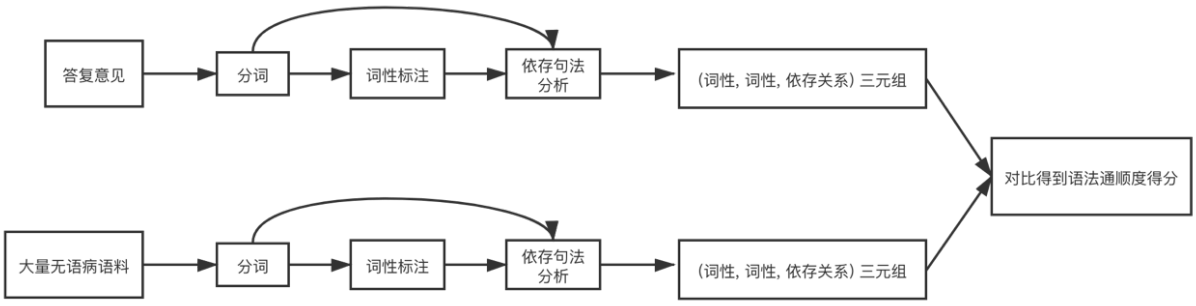

图 17 语法通顺度评分流程

在具体的评分流程（图 17）中，我们首先对于答复意见分词，再对分词进行词性标注，然后再根据分词以及分词的词性进行依存句法分析，得到（词性，词性，依存关系）三元组。

例如在上述例子“经调查，C 市高新区双马中心学校 2016 年前没有收到农村贫困寄宿生补助经费。”中，“收到”的词性是 v，即动词；“双马中心学校”的词性是 n，即名词；则（v，n，SBV）构成一对合法三元组，此时我们再判断该

三元组是否存在于语料三元组（由大量无语病语料生成）中，如果存在则当前三元组语法正确，否则判定为错误。

最后我们根据正确三元组在所有三元组中所占比例得到当前答复的语法通顺度得分，计算过程如公式 28 所示。

$$GrammarScore = \frac{RightTupleNum}{TotalTupleNum} \tag{28}$$

4.3.3 语义通顺度

除了语法之外，语义也是衡量一条答复通顺度的重要维度[10]，具体求解流程如图 18 所示。

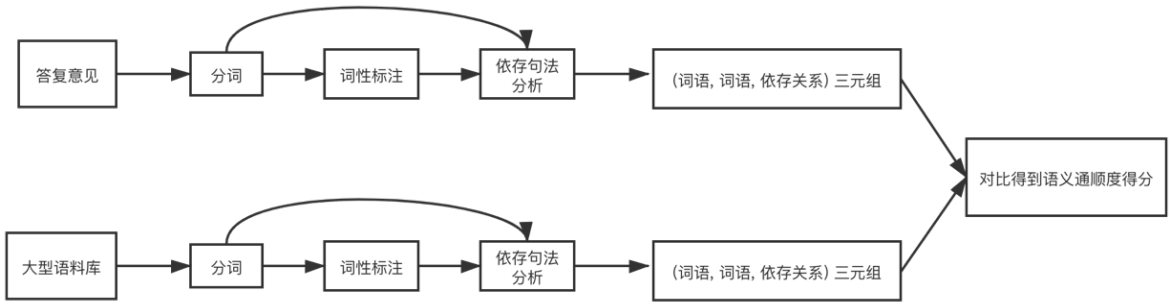

图 18 语义通顺度评分流程

相比于语法通顺度的求解过程，语义通顺度的求解需要将原来的（词性，词性，依存关系）三元组改为（词语，词语，依存关系），例如（v，n，SBV）即改为（收到，双马中心学校，SBV）。除此之外，我们还将原来的无语病语料改为了搜狗实验室公开的大型语料数据库，力求尽可能地涵盖常见的（词语，词语，依存关系）三元组，提高计算准确度。

最后我们根据正确三元组在所有三元组中所占比例得到当前答复的语义通顺度得分，计算过程如公式 29 所示。

$$SemanticScore = \frac{RightTupleNum}{TotalTupleNum} \tag{29}$$

4.3.4 通顺度计算

鉴于语法通顺度与语义通顺度同样重要，并且任意一者得分过低都会强烈影响语句通顺度，即在某一方得分过低，另一方得分却很高的情况下，语句通顺度仍然很低，因此相较平均值算总分，取最小值的方式在此处更为适合。

$$SmoothScore = min(GrammarScore, SemanticScore) \quad (30)$$

我们最终采用公式 30 计算答复通顺度总分，并得到了图 19 所示的数据集中答复通顺度的分布图。

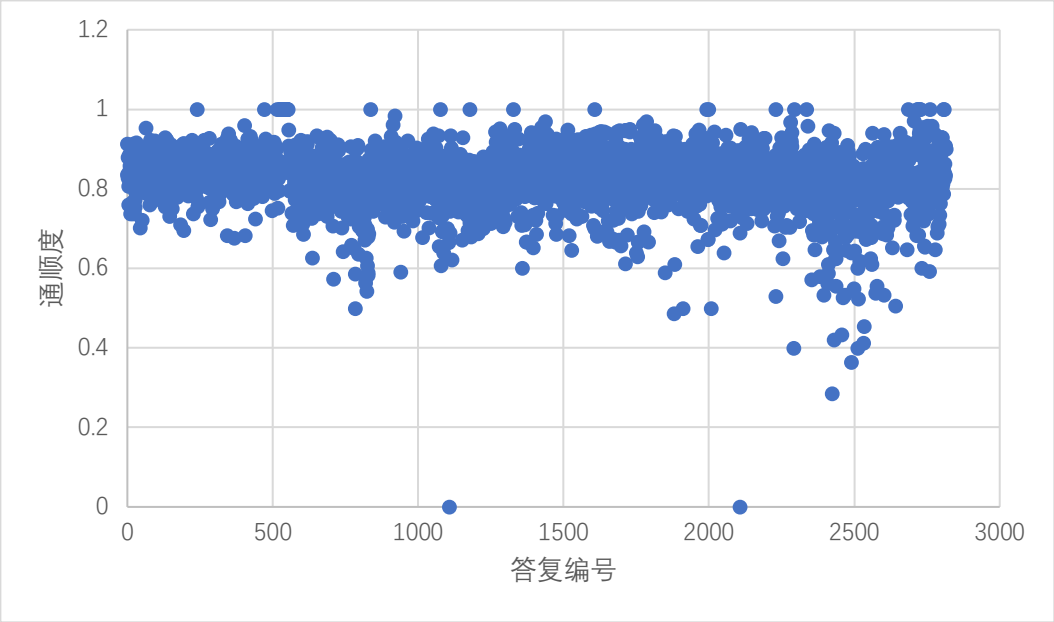

图 19 通顺度分布图

4.4 及时度

因为留言具有时效性，在评价政府答复质量时，我们需要考虑用户发布留言到政府回复这之间的时间间隔，并且考虑到政府部门工作的特殊性，此处的时间间隔仅包含工作日。因此我们定义了一个分段函数来量化政府评价的及时度。

在我们定义的评价体系中，及时度满分为 100 分，该分数会随着工作日的拖延而降低。假设回复间隔了x个工作日，则每天减少的分数如表 4 所描述。

表 4: 回复间隔工作日与分数变化关系

x 的范围（天）	分数
$[0, 10)$	回复及时，100 分
$[10, 16)$	每日减少k分
$[16, 28)$	每日减少$1.1k$分
$[28, 46)$	每日减少1.1^2k分
$[46, 70)$	每日减少1.1^3k分
$[70, 100)$	每日减少1.1^4k分
≥ 100	回复不及时，0 分

最后我们再将及时度（$TimeScore$）的范围从 $[0, 100]$ 映射到 $[0, 1]$ 上，并得到了如图 20 所示的数据集中答复及时度的分布图。

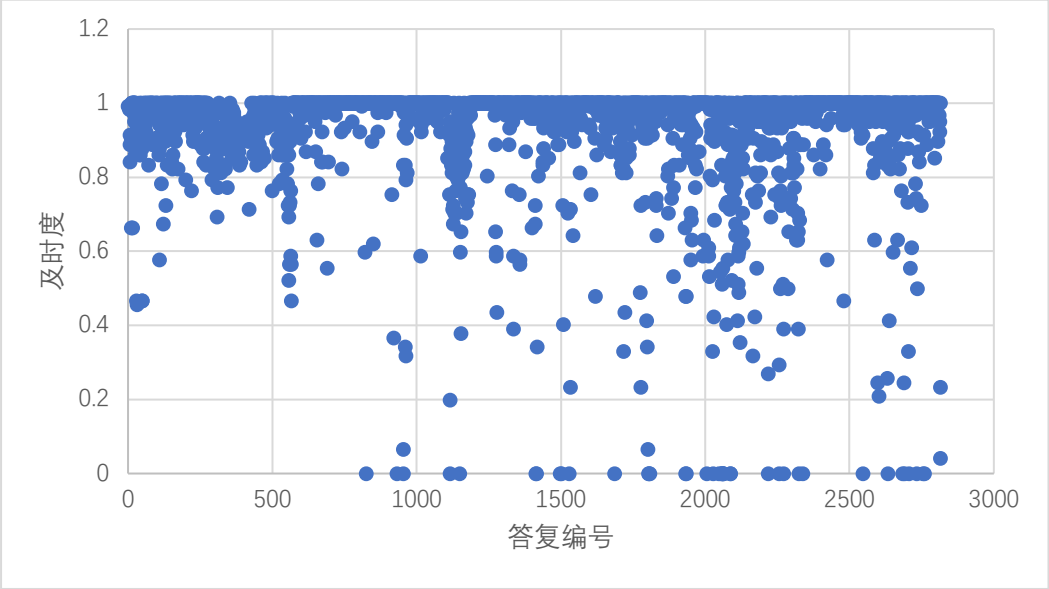

图 20 及时度分布图

4.5 答复总分计算

到目前为止，我们共获得了匹配度（$MatchScore$）、通顺度（$SmoothScore$）、及时度（$TimeScore$）三个指标的对应分数。接下来，我们将根据以下的 3 点考

虑来制定最终的评价方案。

1. 匹配度是三个指标中最重要的

2. 相较于及时度，通顺度更加重要

3. 当匹配度足够低时，通顺度与及时度再高也没有意义

基于上述 3 点考虑，我们决定动态分配三个指标的系数，并将其总和作为最终分数，具体计算如公式 31 所示。

$$TotalScore = a_1 * MatchScore + a_2 * SmoothScore + a_3 * TimeScore \quad (31)$$

其中 $TotalScore$、$MatchScore$、$SmoothScore$、$TimeScore$ 分别指答复的总分以及匹配度、通顺度、及时度得分，各系数 a_1, a_2, a_3 赋值方法如公式 32、33 所示。

$$\begin{cases} a_1 = 0.5 \\ a_2 = 0.3 \\ a_3 = 0.2 \end{cases}, MatchScore \geq 0.3 \quad (32)$$

$$\begin{cases} a_1 = 1 - \frac{5}{3} MatchScore \\ a_2 = \frac{3}{5}(1 - a_1) \\ a_3 = 1 - a_1 - a_2 \end{cases}, MatchScore < 0.3 \quad (33)$$

之所以采用 $MatchScore = 0.3$ 作为分界点，是因为我们发现大量诸如“网友：您好！留言已收悉”这样的万能回复的 $MatchScore$ 在 $[0, 0.3]$ 范围内。我们想要尽可能地降低此类万能回复的得分，因此选取 0.3 作为分界线。

当 $MatchScore < 0.3$ 时，$MatchScore$ 的系数线性递增，占总分的权重线性上升。最后我们将 $TotalScore$ 的范围从 $[0, 1]$ 映射到 $[0, 100]$ 上，并得到了如图 21 所示的数据集中总分的分布图。

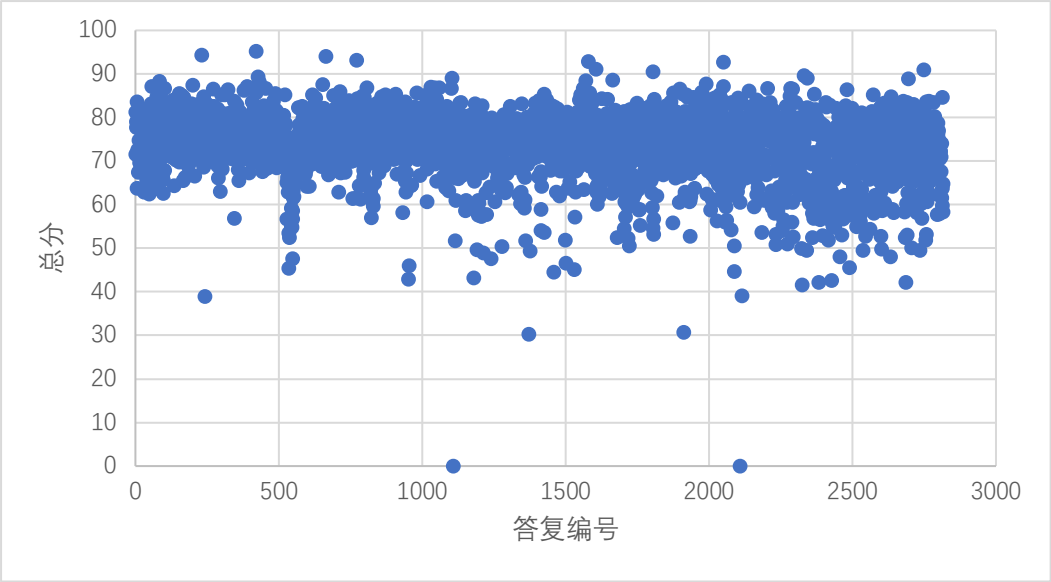

图 21 总分分布图

接下来我们将给出三条分别在匹配度、通顺度、及时度方面具有代表性意义的典型低评分答复，以及一条典型高评分回复，具体留言与答复数据如表 5、表 6 所示。由于留言详情通常较长，因此此处仅列出了留言主题。

首先是第一条答复，属于典型低质量答复。答复地很及时，$TimeScore = 0.96$；但答复内容完全不知所云，$SmoothScore$ 与 $MatchScore$ 均为 0，因此即使回复很及时，依然毫无质量可言。最终使用公式 32、公式 33 所描述的动态分配系数方法进行评分，$TotalScore = 0$，符合实际，具有合理性。

表 5: 典型留言

编号	留言主题	留言时间
1	请求恢复 K8 县县城太平洋-华新书店公交站点	2016/3/11
2	咨询打狂犬役苗报销比例是多少	2018/3/20
3	希望 K 市公交公司公交刷卡异地卡也能享受优惠	2019/1/17
4	关于 A 市实施差别化购房措施的咨询	2018/3/30

接下来再看第二条答复，属于典型的万能答复。答复地很及时，$TimeScore =$

1；语义语法也很通顺，$SmoothScore = 1$；但是内容基本不匹配，$MatchScore = 0.16$，此“两高一低”特征属于万能回复的典型特征。此时如果直接取三者平均值再映射到 $[0,100]$，总分应为 72。但这条答复对留言问题的解决毫无帮助，因此总分及格显然不合理。所以我们采取了上述所描述的动态分配系数的方法，使得最终 $TotalScore = 39$，合理性相较平均值大幅提高，也印证了此种评价方案的合理性。

表 6: 典型答复

编号	答复内容	答复时间
1	0000-00000000　20163	2016/3/31
2	已收悉	2018/3/28
3	根据 2019 年 1 月 17 日，市民在《问政西地省》提出：希望持异地公交卡刷卡在 K 市乘车优惠的问题，我公司与合作方 K 市城市一卡通公司进行了沟通，一卡通回复：目前全国范围内，异地卡在本地乘车均不享有任何优惠政策。	2019/10/23
4	网友“UU0081211”您好！您的留言已收悉。现将有关情况回复如下：根据《关于实施差别化购房措施的通知》，签订拆迁安置协议一年内的属于刚需群体，签订拆迁协议一年后还未购房的不属于刚需群体。感谢您对我们工作的支持、理解与监督！2018 年 4 月 2 日	2018/4/2

再来看第三条答复，属于典型的“迟到”答复。内容较为匹配，$MatchScore = 0.82$；语义语法也较为通顺，$SmoothScore = 0.80$；但是答复时间距离留言时间足足隔了 9 个月，$TimeScore = 0$。此类留言虽然“迟到”，但匹配度与通顺度都很高，也算是积极解决了留言问题，因此打分不宜过低。因此在

我们动态分配系数的评价方法上，最终 $TotalScore = 65$。及格了，符合实际，很合理。

最后是第四条答复，属于典型的高质量答复。内容相关度高，$MatchScore = 0.94$；语义语法也很通顺，$SmoothScore = 0.92$；答复也非常及时，$TimeScore = 1$。答复质量非常高，基本没有明显短板，因此最终 $TotalScore = 95$，合理。

通过上述四条典型答复的分析，不难看出此种动态分配系数的评价方法不但可以压低“万能”、“不知所云”答复的分数，也可以一定程度上地保留高质量但“迟到”的答复的分数，还可以使得高质量、无短板答复凸显出来，符合实际，合理性显著高于常见的均值评价方法。

针对每条答复的各项打分以及总分保存在 T3 留言评价结果.xlsx 中。

5. 总结

本文尝试了三种不同的文本向量化方法在三种分类模型上实现了对于留言的分类，其中 TF-IDF 向量化的支持向量机模型效果最好，F-score 可达 0.92。在此基础上，又使用爬取的政务数据对原数据集 7 类扩充到 13 类，验证了模型的泛化效果，依然较为优秀。在热点问题挖掘上，本文提出的基于 DBSCAN 与点赞数中心化的组合聚类模式能够将热点问题完整、准确的挖掘出来。根据威尔逊得分计算的热度指标可以兼顾留言的数量与点赞数。在答复信息评价方面，本文实现了综合匹配度、通顺度和及时性的评价模型，对每条答复信息给出了科学合理的评分。

参考文献

[1] 贾园园,蔡黎,饶希.网络招聘信息的分析与挖掘[R].湖北:湖北工程学院,2016.

[2] 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.

[3] Song, Yan et al. “Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings.” NAACL-HLT (2018).

[4] 周志华.机器学习[M].清华大学出版社:北京,2016:121.

[5] 马小龙.一种改进的贝叶斯算法在垃圾邮件过滤中的研究[J].计算机应用研究,2012,29(03):1091-1094.

[6] 叶建成. 利用文本挖掘技术进行新闻热点关注问题分析[D].广州大学,2018.

[7] 阮一峰.基于用户投票的排名算法（五）：威尔逊区间[EB/OL].http://www.ruanyifeng.com/blog/2012/03/ranking_algorithm_wilson_score_interval.html,2012-3-20.

[8] 俞霖霖. 面向百度百科的候选答案句抽取研究[D].哈尔滨工业大学,2017.

[9] Wanxiang Che, Zhenghua Li, Ting Liu. LTP: A Chinese Language Technology Platform. In Proceedings of the Coling 2010:Demonstrations. 2010.08, pp13-16, Beijing, China.

[10] 何天文, 王红.基于语义语法分析的中文语句困惑度评价[J].《计算机应用研究》,2017,34(12):5-6.