第八届“泰迪杯”全国数据挖掘挑战赛论文报告

所选题目：C 题

作品名称：“智慧政务”评论的文本挖掘和数据分析

综合评定成绩：_____

评委评语：

评委签名：

“智慧政务”评论的文本挖掘和数据分析

摘要

随着“互联网+”的到来以及科学技术的不断演进，“信息化”，“科技化”逐步成为时代的代名词，政务服务系统也是一样。在逐渐实现微信、微博、网上市长信箱及阳光热线等线上问政，线上沟通的同时，对于市民关心的问题，政府也能够实现积极干预和解决，真正做到服务于民，贴近于民。然而，这也带来了一系列的问题：庞大的文本数据量、冗杂不一的信息渠道以及多人针对同一问题的重复问政，都给仍基本依靠人工进行留言划分和热点整理的政务人员的工作带来了巨大挑战，对于线上市民反馈信息的系统性整理已经迫在眉睫。建立基于自然语言处理技术的智慧政务系统已然是社会治理创新发展的新趋势，这也将会大大提高政府的管理水平和加强政府处理政情民意的效率，做到与时俱进。本文将基于数据挖掘技术，在系统分析整理线上政务留言数据文本的基础上，尝试解决群众留言分类、热点问题整理以及政府答复意见评价等问题，建立一个“智慧政务系统”，提高政府线上问政听政决策的效率和能力。

关键词：评论文本；数据分析；信息提取；文本分类；线上政务

TEXT MINING AND DATA ANALYSIS OF ″SMART GOVERNMENT″ COMMENTS

ABSTRACT

With the arrival of ″Internet +″ and the continuous evolution of science and technology, ″informatization″ and ″technologicalization″ have gradually become the synonym of The Times, and the government service system is the same. In the gradual implementation of WeChat, weibo, the online mayor's mailbox and sunshine hotline online politics, online communication, at the same time, for the public concerns, the government can also achieve active intervention and solution, really serve the people, close to the people. However, it also brings a series of problems: a large amount of text data, mad of different channels of information, and many people on the same problem repeat ask zheng, leave a message to remain basically rely on artificial divisions and hot spot of government affairs personnel brought huge challenges, for the online citizen feedback systematic sorting is imminent. The establishment of intelligent government system based on natural language processing technology has been a new trend of social governance innovation and development, which will also greatly improve the management level of the government and enhance the efficiency of the government in dealing with political conditions and public opinions, so as to keep pace with The Times. Based on the data mining technology, this paper will systematically analyze and sort out the text of online government messages, try to solve the problems such as the classification of public comments, the sorting out of hot issues and the evaluation of the government's comments, establish an

"intelligent government system", and improve the efficiency and ability of the government to ask and listen to government decisions online.

Key words: comment text; Data analysis; Information extraction; Text classification; The online government affairs

目录

1. 挖掘目标

本次建模及方案设计针对政务系统线上留言的数据文本，在进行基本的数据预处理，包括文本去重、数据清洗以及机械压缩去词的基础上，对数据文本进行一级分类，提取有效数据。并利用 python、Excel 等工具，对处理后的数据进行规则合并，建立对应模型，基于各模型的实际效果对数据文本进行针对性分析，并形成政务热点问题以及答复回馈的评价方案，实现线上问政数据的体系化分析，以期解决传统的人工线上留言整理与信息化的庞大线上问政数据量存在的矛盾。

2. 问题 1 分析方法及建模过程：

2.1. 数据预处理

获取到相应附件文本后，我们发现，处理问题 1 的基础数据文件，即附件二所给的数据项冗杂，其中包含许多没有价值的文本数据信息，因此，需要首先对数据进行预处理，提高模型训练的效率，以方便模型的建立。

数据预处理涉及许多工具与方法，我们团队主要运用 python 编程语言进行市民留言，政府答复意见等文本数据的自然语言预处理。包括对可能存在的数据缺失值的处理、数据的规则清洗、数据的文本去重、生僻词词库的引入等处理工作。具体操作流程及方法如下：

2.1.1. 数据集理解

在得到一组基础数据后，进行数据挖掘或者数据预处理前，应该通过对该数据集进行一些规则性提问，以此来明晰这份数据的各类文本特征，并对数据本身是否干净，需不需要作进一步处理的基础问题进行基本判断，判断后方能采取相应的数据预处理工作，增加数据的可用性，以期完成预处理工作的基本目标，提高最终建模的效率。而这些规则性提问主要包括：

> 数据集包含多少数据？
>
> 包含什么字段？字段格式如何？
>
> 字段分别代表什么意义？
>
> 字段之间的关系是什么？可以用作什么分析？能否满足数据本身对分析的要求？
>
> 有没有缺失值？缺失值数量多少？
>
> 现有数据有没有脏数据[①]？
>
> 有没有出现错误名称、空格、拟声词、生僻字等情况？如何处理？
>
> ……

2.1.2. 数据清洗

数据清洗是整个数据分析过程的第一步，是对现有数据进行重新审查和校验的过程，目的在于删除重复信息、纠正存在的错误，并提供数据的一致性。

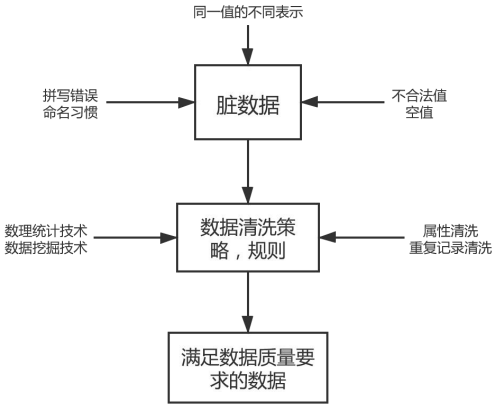

图 1 数据清洗流程示意图

（图片来源：百度百科——“数据清洗”）

数据清洗的第一步首先是对于数据缺失值的发现和处理，我们知道，在现实中，数据缺失是普遍存在的。而造成缺失数据的原因有很多，比如：编辑者人为主观的文本空置、记录设备的不可控问题、网络连接的短暂失效、数据误记、算法本身问题等等。缺失值[②]在文本的表示形式下，就是指空格或者空白的表达，而在 python 语言下，缺失值往往表示为 None、NP.NAN、NAT 等形式。而对于“智慧政务”系统的文本数据而言，也不乏可能出现数据缺失的情况。因此，考虑到

① 脏数据指的是在物理上临时存在过，但在逻辑上不存在的数据。
② *缺失*值是指粗糙*数据*中由于*缺少*信息而造成的*数据*的聚类、分组、删失或截断。

机器处理的效率以及数据本身的挖掘价值，我们通过以下步骤对留言文本以及后续的政府意见答复文本（包含随机及非随机缺失数据）实现了缺失值的处理，处理过程基本包括以下几点：

- 识别缺失数据；
- 检查导致数据缺失的原因；
- 删除包含缺失值的实例或用合理的数值代替（插补）缺失值[1]。

在对数据缺失值进行相应提取、忽略、替换等操作后，数据的完整性得到了保证，在后续的数据清洗工作中，我们对数据集进行了数据类型的调整，实现了格式一致化的操作，将可能存在的标点符号掺杂/大小写不一致/空格重复出现等问题进行了综合考量，实现了数据集的标准化，以便于后续数据处理及模型的建立。

2.1.3. 文本去重

2.1.3.1. 文本去重的简析及原因

文本去重，顾名思义就是删去掉文本数据重复的部分，以求得数据价值的提高。对于文本数据的可视化处理而言，文本去重具有很大的意义，它能够有效减轻机器庞大数据处理量的负荷，并且提高数据处理的效率。针对本题，我们认为进行数据文本去重处理的具体原因如下：

① 针对基于“智慧政务”系统的政务线上反馈，可能存在同一个人为了增加某件事情的热度，以求快速解决而进行多次重复反馈的情况。值得注意的是，这些反馈内容往往相对雷同，文本数据本身的重复率高，容易使用机器辨认。从另一方面来看，同一网络ID对于某一事件的有意多次重复往往会导致事实失真，过于主观，而适度的文本去重则能够有效规避这类情况。

② 开放式的线上问政，进行政务反馈的网民表达能力参差不齐，容易出现高频词、单一表达的语气词以及一些无意义的重复表达，例如：

“真的真的真的”

“好久好久好久”

“一次一次又一次”

这类文本往往重复堆砌，会增加系统后台信息处理的负荷，同时使文本表达过于冗杂，而文本去重能够有效发现这类表达，并解决这类问题。

③ 进行文本数据的归类化工作，需要机器对庞大的数据量进行处理和匹配，数据匹配分类有一定的分词标准，若文本数据的重复性过高，重复的匹配分类，会大大影响数据分类的权重，进而影响最终分类的准确性，文本去重也是进行数据有效分类的基础。

2.1.3.2. 文本去重的具体选择

文本去重有着许多的方法，大多数文本去重算法都是基于对数据相似度的分析，再进行具体去重的操作。文本去重的必要性和效果自然是毋庸置疑的，然而，对于纯文本的数据处理，如果进行过多的文本去重，也会使如数据出现表意不明、轻重不一等问题。因此，针对本次“智慧政务”系统多文本数据的处理，考虑到语义表达的问题，我们团队运用的主要还是文本去重中相对简单的对比删除法，即两两对比，仅删除语句完全相同的数据文本。而对于数据预处理的操作，我们则更多地从其他方面入手，尽量提高数据的可用性。

2.1.4. 生僻字词库[③]引入

生僻字又称冷僻字，意思是不被人们常见或者熟悉的中文字词，包括不同中文字体的呈现形式以及繁简中文的对应形式。

针对“智慧政务”系统的中文文本留言内容，引入生僻字词库是能够帮助机器系统正确分辨识别可能出现的中文生僻字、保证机器运行的进程不受影响、同时促进后续的数据分类及建模的正常运行的基础数据处理操作，同时也是针对中文分词特有的数据预处理工作。合适的生僻字词库引入往往能促进数据的可视化处理，减少数据缺失带来的模型精度下降等问题。

我们团队引入的是常用的中文生僻字词库，也就是网络词典的生僻字词部分，已经能够较好的解决留言文本中难以理解的冷僻字词的输出问题。示例如下：

③ 词库是词语资料的集合，存贮于数据库中以备特定的程序检索调用。

繁	简	繁	简	繁	简	繁	简
舩	船	測	测	並	并	筆	笔
氷	冰	叙	钗	幣	币	錶	表
採	采	參	参	寶	宝	彆	别

表 1 生僻词库 繁简字体对照表（部分）

2.1.5. 数据分词

2.1.5.1. 中文数据分词概述

由于计算机无法直接识别自然语言，我们也就无法直接将预处理后的初始文本扔到分类算法当中得出分类结果。因此我们需要先将文本转化为一定格式的特征编码，文本分类区别于其他分类问题的特点就在于此[2]。显然，转化后的特征编码若能够携带越多的文本特征，就越能帮助分类算法预测出其对应的类别。

中文文本分类最常用的特征提取的方法就是分词。区别于英文天然的存在空格符作为词与词之间的间隔标志，中文文本中词的提取必须通过基于序列预测等方法的分词技术来实现。而分词的概念，指的就是将机器无法理解的中文词语，利用一定规则形成分割标识，也就是常说的词序列，以便于机器识别运算[3]。例如：“今年是鼠年。”能够按照分词效果切分成“今年/是/鼠年/。/”，也能够被切分成“今年/是/鼠/年/。/”，还有很多可能存在的不同分词形式，这就基于不同的中文分词算法结构了。

不同分词算法的运用以及不同分词的结果，都会对最终分类结果以及建模模型造成不可忽视的巨大影响，因此，在运行分词算法、选定分词特征以及界定训练集数据时，必须基于数据进行科学合理的选用，以求得准确的分类结果。

短语	切分
明天就是劳动节	明天/就/是/劳动/节
明天就是	明天/就/是
就是劳动节	就/是/劳动/节
劳动节	劳动/节

表 2 中文分词示例

2.1.5.2. 中文分词具体选用

目前中文分词的方法主要有基于文本匹配的分词算法，基于理解的分词算法以及基于统计的分词算法。本文则主要运用基于 python 的 jieba 库的中文分词法，对 TXT 文档留言数据进行中文分词。jieba 是目前最好的 python 中文分词组件，它主要有以下 3 种特性：

- 支持 3 种分词模式（精确模式、全模式、搜索引擎模式）；
- 支持繁体分词；
- 支持自定义词库

我们运用 jieba 分词首先通过对照典生成了句子的有向无环图，再根据选择模式的不同，依照词典寻找最短路径后对句子进行直接截取。对于未登录到词库（即对照典）的词，我们使用了基于汉字成词能力的 HMM 模型[④]和 Viterbi 算法[⑤]，其大致原理是：采用四个隐含状态，分别表示在单字成词、词组的开头、中间、结尾。通过标注好的分词训练集，和 HMM 的各个参数，最后得到分词结果。

本文主要运用的是全模式下的分词组合。在进行分词文本处理过程中发现，仍存在一些应该作为分词出现的字段没有较好地实现分词结果，例如：“卫生计生”分类标签下，“卫计委”、“抚养费”两词的初步分词结果分别为‘卫’‘计委’ ‘抚’‘养费’。将组合的分词含义拆分，形成的分词结果不尽如人意。因此，我们在词语截取初步结果后加入了 HMM 参数词库进行新词发现以完善分词结果。

HMM，也就是隐马尔科夫模型，是一种基于马尔科夫假设[⑥]的统计模型。之所以为“隐”，是因为相较于马尔科夫过程而言，HMM 有着未知的参数。正如世界上，能看到的事物本身基本都是表象，事物的真正状态往往都隐含在表象之下，并且与表象有着一定的关联关系。从公式的角度上看，我们知道，我们运用各种模型的目的在于针对给出的输入值 X，能够预测出其类别 Y。生成的模型实现对联合概率分布 P（X,Y）的学习，然后通过贝叶斯定理求解其条件概率

[④] HMM 模型，即隐马尔可夫模型（Hidden Markov Model，HMM），是一种基于概率的统计分析模型，用来描述一个系统隐性状态的转移和隐性状态的表现概率。

[⑤] viterbi 算法指的是多步骤每步多选择模型的最优选择问题,在每一步的所有选择都保存了前续所有步骤到当前步骤当前选择的最小总代价(或者最大价值)以及当前代价的情况下前继步骤的选择。

[⑥] *马尔科夫假设*(Markov Assumption):即下一个词的出现仅依赖于它前面的一个或几个词。

$$\arg\max_{Y} P(Y/X)$$

公式 1 贝叶斯定理 求解条件概率

而 HMM 正属于生成模型的有向图 PGM，能够通过联合概率实现建模：

$$P(S,O)=\prod_{t=1}^{n} P(s_t/s_{t-1})P(o_t/s_t)$$

公式 2 HMM 有向图 PGM 联合概率建模

（注：S，O 分别表示状态序列和观测序列）

针对 HMM 的解码及建模模型过程在此不再阐述，本文基于 jieba 分词组件，再配合 HMM 新词发现实现文本留言的分词格式以配合最终的模型实现，具体流程如下：

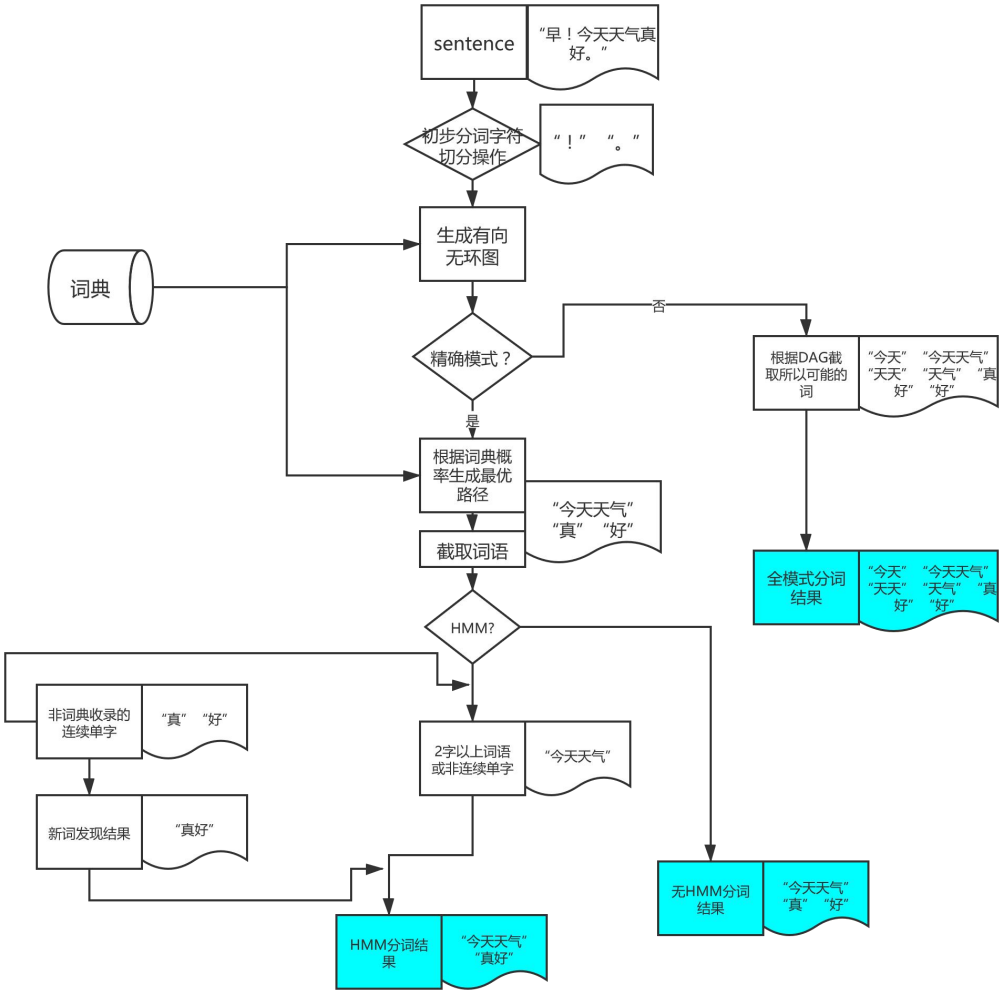

图 2 基于 jieba 组件的文本分词流程

（图片来源：简书—尘嚣看客—“jieba 分词详解”）

2.2. 停用词词库操作

2.2.1. 引用停用词词库

在对留言文本实现中文分词后，留言数据已经成为了一个个分词的集合，能够基本表示为：

$$Z_i = \begin{pmatrix} \varpi_{11} & \cdots & \varpi_{13} \\ \vdots & \ddots & \vdots \\ \varpi_{i1} & \cdots & \varpi_{in} \end{pmatrix}$$

公式 3 中文分词 基本表示

（注：n 代表某条留言文本的分词个数，i 表示留言的总数。）

然而，留言文本中仍然存在着无意义的表达，这类分词表达被称作停用词。停用词有着普遍出现以及无实际含义这两个特征，这样的特征会一定程度上加大模型的运算负荷以及影响分类的词汇匹配准确度[4]。鉴于此，本文引入了中文停用词库，用于对反复出现且无实际含义的，例如“不由得”“不知不觉”“一旦”“一般”和无用的符号等停用词的区分与剔除。

A区 的 卫生 真的 又脏 又乱

A区 卫生 脏 乱

图 3 停用词过滤效果前后对比

2.2.2. 人工查看 review 增加停用词库

在实现停用词词库引入与训练集尝试后，我们发现中文停用词库的停用词并无法完全满足本案例所需的特定停用文本的需求。例如：

我 真的 不知道 为什么 总是 被 骚扰

请 有关部门 及时 整治

图 4 原始数据 未进行停用词操作 文本分词

我 为什么 总是 被 骚扰

请 有关部门 及时 整治

图 5 初步停用词操作 未增加词库 文本分词

于是，我们在人工查看 review 之后，增加了停用词库的词汇，以实现更加有效的停用词筛选剔除，并取得了比较好的结果。

我 被 骚扰

请 有关部门 整治

图 6 增加停用词库 停用词操作后文本分词

2.3. 训练生成词向量

将问政平台的留言的转化为一个机器学习的问题，需要将文本数据训练成词向量，将符号进行数学化处理，常用的文本词向量表达方法就有使用 One-hot 独热编码向量，N 位状态寄存器来对 N 个状态特征进行编码，将一个词映射成一个很长的向量，向量的长度就是词表的长度[5]。例如有两句话，分别是“学习太快乐了，我爱学习”，”我要去学习”，两句话可以得到 10 个不重复的字，那么就决定了在使用向量表示文本时长度为 10。

下列表格第一行表示这两句话构成的字典，对两句话进行表示就需要 2＊10 的矩阵，若是字典中的某个字若出现在句子中，则填写数值为 1，反之为 0。

学	习	太	快	乐	了	我	爱	要	去
1	1	1	1	1	1	1	1	0	0
1	1	0	0	0	0	1	0	1	1

表 3 独热编码示意图

但是这样就存在了不考虑词与词之间的顺序语义问题，同时，在进行留言分类时字典的字词成千上万，那么每条评论转换后的向量维度是数十数百维的，又因为向量中大部分的元素是 0，所以得到的特征将会是离散稀疏的，这就导致了我们还需要采取其他的词向量模型，我们在这里采用 LDA 文档生成主题模型和 Word2Vector 模型。

(1) LDA 文档主题模型

又叫隐含狄利克雷分布，结合贝叶斯公式，将文档集中，每篇文档的主题以概率分布的形式给出，通过分析一些文档抽取出对应的主题分布后，便可以根据主题分布进行主题聚类或文本分类，与此同时，一篇留言可以包含多个主题[6]。

(2) Word2Vector 模型

包含两种词训练模型，分别是 CBOW 模型和 Skip-gram 模型，引入 gensim 模块使用 model.WordVec 进行实现，将整个语料库投入训练，并设置 word 词向量的维度，设置一个句子中当前单词和被预测单词的最大距离，并设置训练模型时使用的线程后进行训练。

2.4. TF-IDF 算法

(1) 算法介绍

TF-IDF（term frequency－inverse document frequency，词频-逆向文件频率）是一种用于信息检索与文本挖掘的常用加权技术。TF-IDF 作为一种统计方法，用以评估一字词对于一个文件集或一个语料库中的其中一份文件的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。简而言之，TF-IDF 算法的核心就是过滤掉常见的词语，保留重要的词语作为文本关键词或特征词。

相比其他词频统计方法，TF-IDF 更容易理解和实现，它的原理就是比较分词在语料库和特定文本数据中分别呈现的频次，用以界定词汇的可用性。一般而言，如果某个分词在留言文本中出现的频率 TF 高，并且在其他文本或语料库中很少出现（即 IDF 低），则认为此词或者短语具有很好的类别区分能力，适合用来分类。TF-IDF 的优势是实现简单、相对容易理解，然而 TF-IDF 算法提取关键词的缺点也很明显，过于依赖语料库，单纯的认为文本频率小的词就越重要，文本频率大的词就越无用，导致算法的精度不是很高，所以应用 TF-IDF 算法也应该作相应的改进。

(2) 算法实现

首先，需要计算的是词语在留言文本中出现的频次，即 IF 的数值。其中，TF 的值越大，说明该词条（分词）出现的频率越大：

$$tf_{ij}=\frac{n_{i,j}}{\sum_k n_{k,j}}，即\ tf_{\varpi}=\frac{在某一类中\varpi词条出现的次数}{该类中所有的词条数目}$$

公式 4 TF 词频计算

其次，需要计算该词语在其他文本或语料库中出现的频次，即逆向文件频率

（IDF）。如果包含词条 t 的文档越少，IDF 越大，则说明词条具有很好的类别区分能力：

$$\mathrm{idf_i}=\log\frac{|D|}{|\{\mathrm{j}:\ \mathrm{t_i}\in \mathrm{d_j}\}|+1}\text{，即 }IDF_{\varpi}=\log\frac{\text{语料库的文档总数}}{\text{包含词条}\varpi\text{的文档数}+1}$$

（注：+1 是为了防止出现分母为 0 的情况）

公式 5 IDF 逆向文件频率计算

最后，依据某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的 TF-IDF，用于关键词分类。

(3) 算法改进

根据留言文本特征的分析，单纯利用 TF-IDF 公式计算也会存在一些问题，包括：

a. TF-IDF 权重大小与特征项的顺序无关，也即特征词在文档中的位置并不影响权重大小。但实际上，就我们的政务留言文本而言，留言主题相比于留言详情，不仅仅具有较高的单文档权重，同时在区分不同留言时也具有较高的贡献度。

b. TF-IDF 公式对留言中出现次数较少的重要人名、地名、职称名等命名实体具有较弱的信息区分能力，而命名实体标识了留言的许多关键信息，对留言话题的识别和演化研究都具有重要意义。有时一个留言文本中还可能出现不同的实例，如时间、地点等，而判断哪一个才是与事件发生直接相关的实体，还需要结合留言时间综合考虑，这也正说明，对于命名实体应赋予更高权重。

c. TF-IDF 认为区分文档能力强的词是在某个留言文本中出现次数足够多，在其他文档中出现次数足够少的词，并不考虑留言的来源。但根据信息化的传播及线上问政的高密度特点而言，一个热点问题必然被不同网民同时问政，且关于该问题的留言的数量和频度都较高，如果一个特征词出现在不同网民留言且于同一时间内均匀出现，即多个留言文本都包含同一个特征项，那么这个特征项对热点话题就具有很大的贡献作用，应该赋予特殊权重。

文本表示模型的最终目标是实现一个有效的特征集，因此权重计算方法必须具备两方面的关键特征，一个是能够代表目标文本内容的完整性，另一个是能将目标文本内容与其他文本相区别的区分性。因此，本文在进行留言热点问题挖掘

时，提出一种基于经典 TF-IDF 的改进算法。

传统 TF-IDF 中 IDF 部分，只考虑了特征词与它出现的文本数之间的关系，而忽略了特征项在一个类别中不同的类别间的分布情况。在某些特殊情况下，如果某词未能出现在语料库里面，则需要考虑将公式平滑为

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

公式 6 TF-IDF 公式平滑改进

特征词	TF-IDF	改进后 TF-IDF
患者	0.001752	0.002061
环保	0.001421	0.001649
噪音	0.001572	0.001644
医疗保险	0.000967	0.001369
污水	0.001182	0.001248

表 4 TF-IDF 与改进型公式的特征权重计算比较（部分）

2.5. 模型评估——“留出法”

一般而言，全体数据集需要划分为互斥的两个数据集合，以便进行魔性的训练和模型的评估，这两个数据集分别为训练集 S 和测试集 T，用数学的方式表达，可以表示为：

$$D = S \cup T, S \cap T = \varnothing$$

公式 7 留出法 数据集划分

在 S 上依据目标需要训练出模型后，用 T 来评估其测试误差，作为对模型泛化误差的评估测试数据。然而，数据集的具体划分标准是什么，各自占比是多少，可能存在的影响因素有什么，这些都是需要思考的。首先，我们在数据集划分时使用的方法是留出法（hold-out），就是直接将数据集 D 划分为两个互斥的训练集 S 和测试集 T 的方法。留出法的核心在于样本平均，或者说是均匀取样。即训练集和测试集两组子集必须从完整数据集合中均匀取样。目的是尽量减少训练集/测试集与完整数据集合之间的偏差，一般的做法是进行随机取样，在样本数量足够大时，就能达到样本平均的效果。因此，我们应该明晰运用留出法进行数据

集划分时需要注意的一些问题：

1) 测试/训练集的划分需要注意数据分布的一致性，即不能使数据集的划分出现不必要的额外偏差，造成数据污染，应该减少数据分集对模型结果可能产生的影响。
2) 单次使用留出法得到的结果往往具有偶然性，可能存在较大误差。因此在使用留出法时，我们需要注意对数据集进行若干次随机划分，重复进行实验评估并取平均值作为留出法的评估结果。
3) 划分训练/测试集的初衷，是为了评估出数据集的模型准确度，或者说精度。但对于两个集合包含的数据量分配，却存在着两难的境地：若训练集 S 数据量大于测试集 T，则测试集可能存在因数据量不足而导致的评估结果不准。若测试集 T 数据量大于训练集 S，则被训练的模型的分类准确度就得不到保障，同时评估出来的结果可能失去保真性（fidelity）

综合上述问题，我们选择划分包含 70%样本的训练集和 30%的样本测试集用于作留出法评估，以保证训练出来的模型分类的准确度。

2.6.XGBoost 处理

2.6.1. XGBoost 概述

XGBoost 是一种集成学习算法，属于 boosting 算法类别，和 GBDT 算法模型一样，它也是一个加法模型，换句话说，XGBoost 是属于梯度提升树，也就是 GBDT 这个范畴之内的，其基本思想原理也是逐颗树进行学习，利用之前模型的残差进行多次迭代拟合，形成梯度学习，最终实现分类。

2.6.2. XGBoost 的特征值选择及分类

XGBoost 主要使用的是 levelwise 策略，即每次对同一层级的全部叶子结点尝试进行分类。运用二叉树的数据结构，将全部留言样本数据都放在同一个叶子节点上，叶子结点以此不断通过二分裂，逐渐生成一颗树。根据树的剪枝策略的不同，会形成不同的分裂方式，我们的 XGBoost 主要运用的是预剪枝策略，即仅当分裂后形成的目标函数发生下降时，才会进行分裂。

和单纯的 GBDT 模型算法不同的是，XGBoost 采用特征并行的方法选择要进行分类的特征，即多个线程，并尝试把各个特征同时作为分裂的特征，找到最优分割点，借二阶泰勒展开寻找新的目标函数，计算对比效益之后，选择增益最大的作为分裂特征。

XGBoost 基于残差优化的算法，建立多个回归树，使得鼠群的预测值尽量接近真实值，并且具有较大的泛化能力，在实现特征值选择及文本分类上有着更可观的训练结果。

2.6.3. XGBoost 算法参数优化

构造一个使用 XGBoost 的模型十分简单。但是，提高这个模型的表现就有些困难。XGBoost 算法的参数较多，想要提高模型的泛化能力，优化模型参数也是必不可少的一个步骤。下面我们详解我们在运用 XGBoost 模型时使用的参数优化方法，需要进行如下步骤：

1) 采用树的结构运行数据，其中 gblinear 基于线性模型，booster：gbtree。
2) 因为目标函数使用 Xgboost 做多分类问题，所以需要设置 num_class。Objective：multi:softmax。
3) 定义 learning rate 和参数优化的估计器的数量，max_depth=6:，一般而言，该取值在 3-10 之间属于正常参数，值越大则越复杂，我们选择起始值为 6，以构造树的最大深度。
4) min_child_weight=1.2：此参数根据数据集的平衡性进行调节，表示一个子集所有观察值的最小权重和，即某些叶子节点下的值会比较小，又称最小叶子节点。
5) Gamma=0.2：此参数是用于控制是否剪枝的参数,只有损失函数下降超过这个值节点才会越大越保守。这个参数需要在后期进行适当调整。
6) Lambda=4，以此参数作为控制模型复杂度的权重值的 L2 正则化项参数，参数越大，模型越不容易过拟合。
7) Subsample=0.4，即每棵树随机采样 30%的训练样本，该值通常取 0.5-1。
8) colsample_bytree=0.5，表示生成树时进行的列采样率，特征采样率。
9) Silent=0：该参数设置成 1 则没有运行信息输出，因此为防止出现上述情况，

我们将该值设置为 0。

10) Eta=0.007：更新中收缩步长，在每次提升计算之后，算法会直接获得新特征的权重。 eta 通过缩减特征的权重使提升计算过程更加保守

11) Nthread=4：更新线程数，缺省值是当前可获得的最大线程数

12) scale_pos_weight = 1: 这个值是因为类别十分不平衡。

2.7. GBDT 分类处理

2.7.1. GBDT 概述及基本原理

GBDT 全称梯度下降树，是在传统机器学习算法里对真实分布拟合的最好的几种算法之一。我们知道，GBDT 是通过采用加法模型（即奇函数的线性组合），以及不断减小训练产生的残差来达到将数据分类或回归的算法。具体地说，GBDT 算法主要是通过多轮迭代，每轮迭代产生一个弱分类器，每个分类器在上一轮分类器的残差基础上进行训练。在弱分类器选择上一般选择的是分类回归树，这也就限制了分类器的深度，降低了分类器的复杂程度。最终的分类器，也是在每轮训练得到的弱分类器加权求和得到的。

2.7.2. GBDT 应用一级分类问题

GBDT 应用分类问题时使用的仍然是 CART 回归树，因为应用分类树作为弱分类器的话，不同类别下输出的不同样本相减其实是没有意义的。GBDT 应用于分类，会经过什么流程呢？首先，在进行第一步训练时，我们针对留言样本 x 任何可能存在的 7 个一级分类指标的类都训练出一个分类回归树。例如，样本 x 属于“环境保护”指标下的类，那么顺序来看就是属于第二类，我们就可以用一个多维向量[0,1,0,0,0,0,0]来进行表示，其中 0 表示该样本不在该类，1 表示样本属于该类。

针对一共存在的 7 个分类标签即 7 个类的情况，我们实质上是在每轮的训练中同时训练 7 颗树，利用生成 CATR TREE 的特定程序解出这 7 颗树，再仿照多分类的逻辑回归，产生相应概率，针对不同类别形成相应残差，最后进行第二轮训练，重新构造 7 颗树，一直迭代 M 轮，完成训练。

在训练完成之后，针对进行分类的新样本 Y，就可以形成相应的 7 个式子对应 7 个值，由此生成该样本 Y 可能存在某个分类标签下的概率，进行对比最后生成分类结果。

2.8. SVM 支持向量机文本处理

2.8.1. SVM 简介

支持向量机，因其英文名为 support vector machine，故一般简称 SVM，通俗来讲，它是一种二类分类模型，其基本模型定义为特征空间上的间隔最大的线性分类器，其学习策略便是间隔最大化，最终可转化为一个凸二次规划问题的求解[7]。对于文本分类而言，支持向量机也是一种公认的分类效果较为占优的方法，它是一种建立在统计学习理论基础上的机器学习方法，有效解决了需要无穷大样本数量的问题，它只需把一定文本经过计算抽象成向量化的训练文本数据，就能够有效提高分类的精确率。

2.8.2. 基于 SVM 的文本分类过程

基于 SVM 支持向量机的文本分类算法主要有四个方面的内容，即文本特征提取、文本特征表示、归一化处理和文本分类[8]。

1. 文本特征提取

文本特征提取的基本方法是根据文本中词汇的特征向量，通过设置特征阀值的办法选择最佳特征作为文本特征子集，建立特征模型。而特征项的提取步骤也可以基本总结为：

（1）对全部训练文档进行分词，由这些词作为向量的维数来表示文本；

（2）统计每一类内文档所有出现的词语及其频率，然后过滤，剔除停用词和单字词；

（3）统计每一类内出现词语的总词频，并取其中的若干个频率最高的词汇作为这一类别的特征词集；

（4）去除每一类别中都出现的词，合并所有类别的特征词集，形成总特征词集。最后所得到的特征词集就是我们用到的特征集合，再用该集合去筛选测试集

中的特征。

在上文的文本预处理操作中，我们基本按照 SVM 向量机的特征项提取，实现了文本分词、剔除停用词，选取特征词等方法，实现了文本的特征提取。

2. 文本特征表示

文本特征表示，针对本题文本分类的初衷，其实就是上述文本特征进行词频统计，并利用 TF-IDF 算法进行特征表示。

我们知道，在对数据文本进行各项处理后，我们已经得到了较为干净的数据集，针对数据所需要的关乎案例的分析处理也能够进行了。对于一级分类的实现，有一个很简单的思路，就是找寻分类标签下各特征词出现频次最高的词汇，再与各分类标签进行匹配对应。而这个想法的关键就在于对某些特定词汇出现频率的统计，也就是词频的统计。不难理解，出现次数最多的就是“着”、“呢”“的”这一类最常用的词，它们也正是“停用词”，在 13③2.2 停用词词库操作一栏我们已经进行了有效的过滤，留下了能够有效进行词频统计的有意义的数据。

即使如此，也不能说明数据就能够直接进行效果很好的词频统计以确定分类了。值得注意的是，针对某些表意不一的词汇，仅依靠它出现的频率并无法准确说明它的分类特征。如：在进行词频统计时，我们可能会在留言文本中发现“政府”“停电”“施工”这三个文本分词的词频是一致的，但这能说明他们的重要性一致吗？或者说这能表示他们的标签特征吗？显然答案是否定的。

相反，我们却有理由认为“停电”“施工”两分词的重要程度要高于“政府”。因为是向政府留言，“政府”一词出现的概率本身就很高，这样也就使基于词频统计的分类指标遇到了难题。由此我们知道，我们需要的是一个重要性的调整系数，用来衡量一个词是不是常见词。即如果某个词比较少见，但是它在这段文本中多次出现，那么它很可能就反映了这篇文章的特性，正是我们所需要的关键词。用统计学语言表达，就是在词频的基础上，要对每个词分配一个"重要性"权重。对于常见的词（"政府"“有关部门”等）给予较小的权重，较少见的且有较大特征性的词（"停电"、"施工"等）给予较大的权重。这个权重也叫做"逆文档频率[⑦]"（Inverse Document Frequency，缩写为 IDF），它的大小与一个词的常见程度成反比。因此，在运用 SVM 进行文本特征表示时，我们主要运用的也是 TF-IDF

⑦ 逆向文件频率（IDF）：某一特定词语的 IDF，可以由总文件数目除以包含该词语的文件的数目，再将得到的商取对数得到。

的算法模型。

3. 文本归一化处理

文本的归一化处理，指的是依据各类算法实现文本词频的精确化处理，仅仅依靠文本数据得出的文本词频，直接应用于文本分类，往往会存在较大误差，而利用文本归一化处理就能较好的规避这些误差，笔者认为，文本归一化处理有以下两点优势。第一个就是提升模型的收敛速度，数据在进行归一化后，最优解的寻优过程明显会变得平缓，更容易正确的收敛到最优解。

第二个优势是归一化处理能够有效提升模型的精确度。我们知道，在多指标评价体系中，由于各评价指标的性质不同，通常具有不同的量纲和数量级。当各指标间的水平相差很大时，如果直接用原始指标值进行分析，就会突出数值较高的指标在综合分析中的作用，相对削弱数值水平较低指标的作用[9]。因此，为了保证结果的可靠性，就需要对原始指标数据进行一定的处理。针对本题，在实现文本分类的目标驱动下，我们需要予以不同的分类指标实现可能存在的不同维度的分类，因此，我们更需要将数据进行有效的归一化、标准化处理，目的是让不同维度之间的特征在数值上有一定比较性，提高分类器的准确性。

4. 文本分类实现

政务留言文本数据经过文本预处理、特征提取、特征表示、归一化处理后，已经成功抽象形成了向量化的样本集，接下来需要实现的就是进行样本集与训练好的模板文件的相似度计算，将与训练好的模板文件相似度符合标准的样本集纳入该分类，不属于的列入下一模板文件分类进行计算，直到寻找到相应的分类集，最终实现政务文本的一级分类。

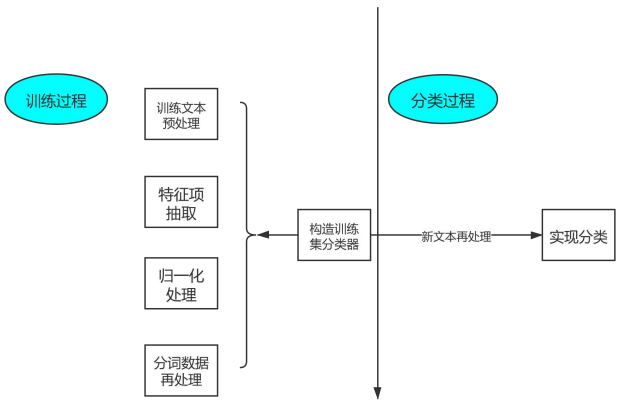

图 7 SVM 特征向量机 文本分类 流程

2.9. 模型评估

本次建模评估分别有以下四个指标：

a. 分类准确率 Precision。

类别 c_i 的分类准确率 ρ_i，如 所示。

$$\rho_i = \frac{\text{分类结果中正确分为}c_i\text{的样本个数}}{\text{分类结果中所有分为}c_i\text{的样本个数}}$$

公式 8 分类准确率 具体算法

b. 分类召回率 Recall。

类别 c_i 的分类召回率 r_i，如 所示。

$$r_i = \frac{\text{分类结果中正确分为}c_i\text{的样本个数}}{\text{类别为}c_i\text{的实际样本个数}}$$

公式 9 分类召回率 具体算法

c. F1 分数 F1-score。

类别 c_i 的 F1 分数 $f1_i$，如式 所示。

$$f1_i = \frac{2 \times \rho_i \times r_i}{\rho_i + r_i}$$

公式 10 F1-score 具体算法

将全部数据进行不同类型模型建模后得出，三种不同模型算法得出的评估结果不尽相同，详情见下表：

	XGBoost			GDBT			SVM		
参数	P	R	F1	P	R	F1	P	R	F1
城乡规划	0.76	0.50	0.60	0.78	0.28	0.42	0.59	0.43	0.50
环境保护	0.84	0.90	0.87	0.75	0.90	0.82	0.76	0.81	0.78
交通运输	0.84	0.80	0.82	0.84	0.58	0.68	0.63	0.71	0.67
教育文体	0.74	0.68	0.71	0.77	0.58	0.66	0.66	0.40	0.50
劳动与社会保障	0.76	0.86	0.80	0.64	0.91	0.75	0.65	0.75	0.70
商贸旅游	0.89	0.90	0.90	0.91	0.89	0.90	0.86	0.88	0.87
卫生计生	0.89	0.81	0.85	0.96	0.72	0.82	0.82	0.85	0.83
F1-score	0.8165			0.7671			0.7258		

表 5 三模型 评估指数 对比表

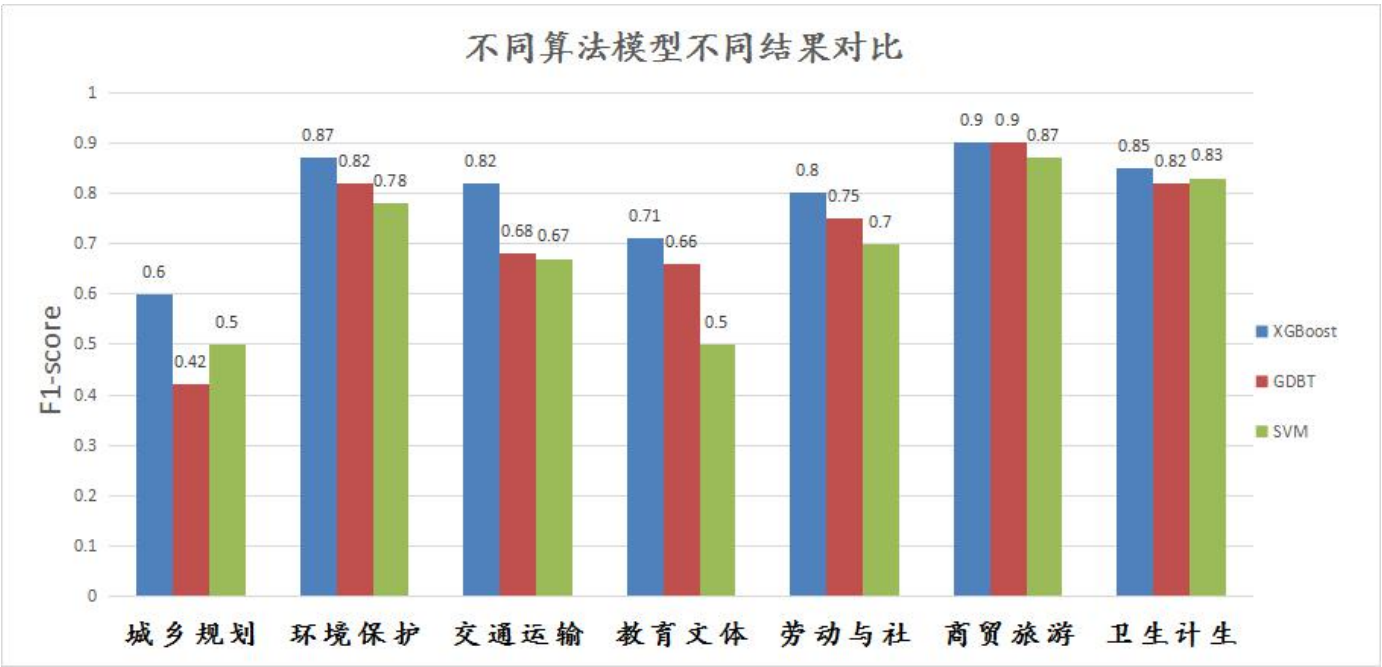

图 8 三模型 F1-score 不同结果对比图

2.10. 模型优化

2.10.1. TF-IDF 改进

在前文 TF-IDF 算法中，我们已经提及对于原始 TF-IDF 算法的平滑改进，鉴于该算法我们普遍应用于上述三个模型之中，所以我们将对该算法的具体改进思路流程置于模型优化的栏目中，具体如下。

对于传统 TF-IDF，一些生僻词的 IDF(反文档频率)会比较高，如果一个特征关键词只在某一个一级标签中的个别文本中大量出现，在其他一级标签类内的其他大部分文本中出现的很少，那么可能会将这些个别文本当作是该一级标签类中的特例情况，不认为该特征关键词具有代表性。，所以，如果考虑将词在文档内的词频率改为在对应的一级标签内的词频率可进行提高词频计算的精度。

2.10.2. 为避免过拟合进行的模型改进

在实验结果可以发现 XGBoost 准确率较于其他两种模型更优，然而模型仍有一定的提升空间，为了避免过拟合的情况在调参时过于保守，可以进一步调优正

则化参数，降低模型的复杂度，提高模型的泛化能力和表现。同时对于模型的 learning rate 学习率，可以结合 cv 函数在每一次迭代中使用交叉验证法，根据返回值选择对用该学习率的理想决策树数量

对于 GBDT，可以引入划分最小不纯度参数，限制决策树的增长，如果节点的不纯度小于该阈值则不再生成子节点直接作为叶子节点。对于 SVM 模型，选用的 kernel 核函数是线性的，适用于线性克风特征数量多的数据集，但是对于政府留言的多分类模型，数据的分布不一定是线性的，可以进一步明确数据之间的关系，选择更为适用的核参数，同时在合理的范围内调优惩罚因子 C 优化分类的误差，总体来说各模型仍有较大的提升空间。

3. 问题 2 分析方法及过程：

3.1. 指标体系构建及理论依据

3.1.1. 解法概述

经过对政务留言文本的系统分析，我们团队将题目 2 的热点问题挖掘分为了两个环节，即热点事件确定+表格数据匹配。热点事件确定是指基于一定的数据结构基础，将已经在模型中经过演算形成文本分词的政务留言，运用 TF-IDF 算法提取分词中能实际概括留言文本内容的核心词，再经突发检测算法实现，提取出留言分词中相对增长率突然增加的核心词，以此确定该核心词在文档中的突发权重指数，最后经文本聚类确定焦点词及其相应突发权重系数作为热点事件确定的依据。第二步则将文档中的点赞数及反对数有效利用，作为下一级的分类标准，针对最终热度分一致或近似的文本分词，再经过时效性指标最终比较，确定热度优先级。表格数据匹配部分则是基于热点事件，匹配其具体留言文本，实现表格数据配对制作，问题解决。

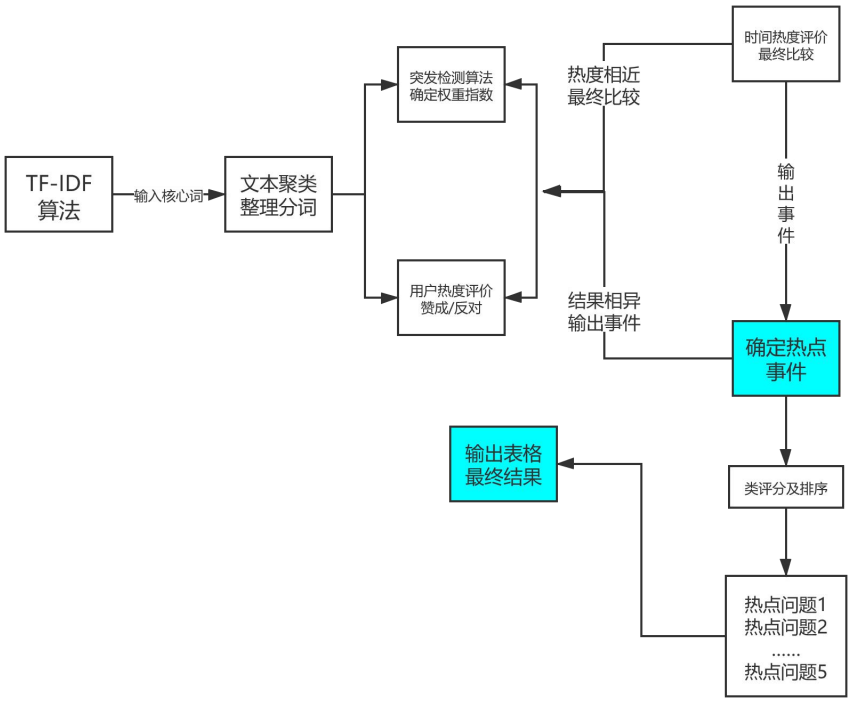

图 9 热点问题挖掘处理方案 流程图

3.1.2. 热点事件挖掘

3.1.2.1. 思路分析

对于基于大量文本数据的热点问题挖掘，其核心在于明晰聚类指标及确定最终系数权重。首先，针对本问题第一步要求的将某一时段内反映特定地点或特定人群留言的归类问题，需要明晰任务的先后顺序及重点所在。对于极度差异化的地点及人群，可能在不同文本留言中各不相同，也大概率存在对于同一地点和人群的不同文本表达。例如：

“A 小区 XXX 村”

“B 小区 隔壁 A 小区 XXX 村”

“A 小区 XXX 路口 村子”

以上表达，他们可能讲的是同一地点，只是文字表达的各不相同，对于人群及时间也是这样。这些差异化的表达及客观存在的多种多样的人群及区域，都会对问题算法及最终聚类造成很大的不确定性。同时，我们在对问题进行分析的过程中不难得出，无论是特定地点、特定人群亦或是特定时间段，最终服务的都是这一特定的热点事件。因此，我们团队决定从事件提取的角度出发，先求得热点

事件，再基于事件的先后排位决定最终特定人群、地点、时间段的表达，最终得出匹配的事件及对应的特定留言信息，减少算法的数据运算量，提高效率。

3.1.2.2. 热度指标

网络政务留言的热度问题本质上是指一个留言或一个事件引起的民众关注及激起讨论的热烈程度，如何定量地给予文本数据热度，目前学术界也没有标准的指标体系[10]。然而，笔者认为，基于这样的实际情况，有针对性、有依据地构造热度指标将会是对于热度事件确定而言最有效的方法。那么，影响留言文本热度的因素有什么呢，对于此，经过对留言文本界面的分析，我们得出了四个分析角度，即分类标签+数据本身+用户分析+时间热度。

A. 分类标签

不难发现，在对题目一的分词文本进行分类处理时得到的规范化的数据和分类模型，对于本题而言也是有很大启示作用的。在数据量方面，我们计算发现，一级分类标签中“劳动与社会保障”标签包含的数据数量最大，由此我们得到启发：不同分类标签下的文本，其初始文本热度不应该一致。例如：涉及劳动民生的“劳动与社会保障”标签本身受关注程度就高过其他分类标签，在分类标签中其实也包含着热度的标准。因此，如果还是给每条留言文本同样的热度就显得不贴合实际了，我们的解决办法，就是将分类标签数据量作为热度分的影响因素，植入一个权重系数，在计算初始热度时进行整合。

B. 数据本身

所谓数据本身，指的就是留言文本分词化的呈现结果，针对数据本身，我们能做到的是实现词语的数据化处理，那么，基于这些可能的数据结果我们又能实现什么呢？我们得出的答案是实现词汇的热度定义，因为，我们认为，目标热度事件在编程化思想下，其实也就是分词化、向量化的呈现结果，如果留言文本分词能够经过数据处理实现各核心词汇的权重指数划定，那基于这样的焦点词，我们就能依据各焦点词汇的突发权重指数大致匹配出热点事件分词，实现任务匹配。

在经过思路分析后，我们得出，这一任务关键在于核心词汇及词汇突发权重指数的确定。在核心词汇的确定上，我们运用的是 IF-IDF 算法，即把文本中出现词频较高且总体中词频较低的词汇统计聚类，形成每一文本对应的核心词，这些

核心词能够基本表达留言文本的实际内容，故称作核心词。

当然，在对数据文本运用 TF-IDF 算法计算核心词以确定热度权重依据时，我们也应该考虑到 该算法在进行热度指标计算时会存在的一些问题，考虑是否运用基于经典 TF-IDF 算法改进的更具有文本区别度的改进算法。经典 TF-IDF 算法存在的不足见详见 16(3)算法改进。

在确定核心词后，我们需要对核心词再做一次突发权重指数确定，在这里我们用的是克林伯格于 2002 年提出的 Burst Detection(突发检测)算法，该算法能够有效计算一段时期内相对增长率突然增加的焦点词在文档流中的突发权重指数[11]，简单的说，就是能够实现核心词基于其他文本的相对增长率计算，以此为基础为核心词匹配突发权重指数，在本题中，该指数越大，说明相应的核心词热度越大。经过以上操作，我们成功实现了核心词及权重指数的确定，完成了热度本身角度的指标。

C. 用户分析

政务留言文本不同于其他网络平台留言的是，它是纯粹的文字表达，并不包含基于不用用户群体或者不同用户做法的用户模型分析，因此，在政务留言文本中，我们仅选取了参考价值较大的用户留言互动分析，即不同用户针对已有留言文本的点赞数与反对数。这个数据能够通过数字直观清晰地表达用户对于某一事件的主观印象，这自然也是影响事件热度的一大指标

D. 时间热度

判断一件事情是不是热门，是不是民众所关心的大事情，还有一个指标依据就是时间，也可称作时限。通俗的说，一个事件在大众的视野里久久没有消失，或者说持续备受关注，那么无疑这件事情就应该是热度比较高的事件了。因此，我们认为，时间指标能够成为左右时间热度的一大指标。但值得注意的是，对于新闻媒体方面来说，时间对事件热度的影响却是与我们恰恰相反的，在新闻媒体的角度上看，一个事件如果历时很短，但讨论数很大，那么它无疑是一个热度很高的新闻，换种方式来看，一个事件如果历时很长，讨论数也很大，它却不一定能够作为热度较高的新闻。也就是说，在新闻媒体行业，对于热点事件而言，如果事件发生的时间更为集中，则说明该事件更有新闻意义上的热度，但在我们的文本分析中却不是这样，这就存在一定歧义。基于此，我们决定将时间热度仅仅

作为基于前两个指标形成的热度评价接近或一致的事件的最终评价标准。例如：

A 事件与 B 事件在焦点词词频和用户互动指标上评分接近，那么就可以通过进行时间热度的最终比较，若 A 事件持续的讨论时间更长，那么我们就有理由认为 A 事件更是人民所关注的，与他们生活关系更紧密的热点事件，B 反之亦然。

3.1.2.3. 最终评分

基于上述算法及指标分析，目前我们已经顺利得出每个留言文本的标签数据量、不同标签不同分词词频多少的排位、不同文本对应的焦点词、可以作为热度依据的突发检测权重指数、各文本的点赞数/反对数，以及可能存在的最终时间段长度比较，因此，我们需要对以上多个指标予以权重实现最终评分公式的计算，以求每一条留言文本对应的热度分并得出排位前五的热点事件。

基于此，我们基本将热度分划分为三个模块的计算，即：

$$P_{\chi} = P_{\alpha} + P_{\beta}$$
$$P_{\alpha} = \mathrm{k}\sum A_{\mathrm{i}} P_{\mathrm{i}} (\mathrm{i} = 1、2)$$
$$P_{\beta} = 0.01 \times \sum \frac{P_{\mathrm{i}}}{\mathrm{i}} (\mathrm{d}^{+} - \mathrm{d}^{-}) (\mathrm{i} = 1、2)$$
$$P_{\mathrm{t}} = T_{\mathrm{n}} - T_{\mathrm{l}}$$

公式 11 热度评价算法

注：χ 表示最终热度分，α 表示初始热度分，β 表示用户互动热度分，P_{t} 表示时间热度分，k 表示某分类标签对应的初始权重系数，A_{i} 表示对应文本第 i 个焦点词出现频次，P_{i} 表示第 i 个焦点词的突发权重指数，d^{+}、d^{-} 分别表示对应文本的赞成数和反对数，T_{n}，T_{l} 分别表示同一事件最后一次留言时间和第一次留言时间

① 初始热度分

初始热度分有两个组成部分，一个是分类标签下的初始热度分，另一个是数据本身的初识热度分。首先是分类标签角度，对于具有不同数据量的分类标签，基于不同的基础权重作为积数，计入初始热度分。具体如下表：

数据标签	总数据量	权重基数
城乡建设	613	0.7
环境保护	1969	2.1
交通运输	877	1.0
教育文体	1215	1.3
劳动与社会保障	2009	2.2
商贸旅游	1589	1.7
卫生计生	938	1.1

表 6 不同数据标签 权重基数

初始热度分的第二个环节是基于数据本身的角度，主要是依据焦点词词频及其突发权重指数进行计算的。鉴于每个文本的长短不一，分词表达也存在较大差异，我们选择对每个文本提取 2 个突发权重指数相对较高的焦点词，并分别将其词频与突发权重指数相乘得到文本的初始热度分。因为基于数据处理得到的结果更加客观可信，所以初始热度分在热度评价优先级中为第 1 级，即在热度分计算时首先进行的就是对文本的初识热度分的计算。

例如，某留言文本 w 属于“环境保护”分类标签，文本中突发检测指数较高的两个焦点词分别为“臭气”和“污染”，其对应词频 A_{i} 分别为 5 和 6，检测指数 P_{i} 分别为 4.8 和 1.2，那么该 w 文本的初始热度计算如下：

$$P_{\alpha}=\mathrm{k}\sum A_{\mathrm{i}}P_{\mathrm{i}}(\mathrm{i}=1、2)$$

$$P_{\alpha(\mathrm{w})}=\mathrm{k}\sum A_{\mathrm{i}(\mathrm{w})}P_{\mathrm{i}(\mathrm{w})}=2.1\times(5\times4.8+6\times1.2)=65.52$$

② 用户互动热度分

我们知道，在政务留言文本中有一个选项，就是对其他用户的留言进行赞同/反对的投票，这自然也将成为我们热度分评价的一大指标。因此，基于用户热度分的计算公式，我们对各个文本的赞成数和反对数进行求和，最终结果再乘以其对应文本的 2 个焦点词的突发权重指数的平均数，得出最终的用户互动热度分。乘以指数平均值的目的是给予用户互动热度分一个与初始热度分评价近似的权重，保证热度评价的科学性。另外，我们在查看数据时发现留言建议之间点赞数/反对数存在较大差异，为避免点赞数/反对数对数据指标评分造成结果偏倚的

极端影响，我们在用户互动热度分中引出权重系数 0.01，以求评分指标的相对平衡。用户互动热度分评价是热度评价优先级中的第 2 级，也就是进行评价的第 2 步。

例如，还是某留言文本 w，其点赞数和反对数分别为 4 和 1，那么其用户互动热度分则为

$$P_{\beta}=0.01\sum\frac{P_{\mathrm{i}}}{\mathrm{i}}\left(\mathrm{d}^{+}-\mathrm{d}^{-}\right)$$

$$P_{\beta(\mathrm{w})}=0.01\times\sum\frac{P_{\mathrm{i}(\mathrm{w})}}{\mathrm{i}_{(\mathrm{w})}}\left(\mathrm{d}_{\mathrm{w}}{}^{+}-\mathrm{d}_{\mathrm{w}}{}^{-}\right)=0.01\times\frac{4.8+1.2}{2}(4-1)=0.09$$

③ 时间热度分

基本上依靠这两步计算，我们已经能依据突发权重指数和赞成/反对数对文本进行热度排优了。但本着科学的原则和可能产生的偶然性情况，我们认为可能会存在前两步计算后得分相同或近似的留言文本。基于此，我们将会对这类文本进行第 3 步评价，即时间热度评价，将文本阐述事件的时间段区间作对比，以天为单位，实现最终的热度对比，延续时间长的文本事件，将优先作为热度事件，以此作为最终结果确定热度排位前 5 的热度事件。

例如，相对于某留言文本 w，有另一留言文本 g，其焦点词分别为“违建”和“噪音”，属于“城乡建设”标签下，其初始热度分和用户互动热度分之和为 66，与 w 文本热度分相近，因此，我们有必要针对这两类文本进行时间热度比较。假设 w 文本最后反馈时间和最初反馈时间分别为 2017.12.29 和 2017.12.14，而 g 文本分别为 2018.9.1 和 2018.8.25，则他们的时间热度分计算如下：

$$P_{\mathrm{t}}=T_{\mathrm{n}}-T_{\mathrm{l}}$$

$$P_{\mathrm{t}(\mathrm{w})}=T_{\mathrm{n}(w)}-T_{\mathrm{l}(w)}=15$$

$$P_{t(g)}=T_{n(g)}-T_{\mathrm{l}(g)}=6$$

很显然 15>6，根据热度分的算法规则，我们确定 g 文本优先作为热度占优的留言文本。

④ 热度评价结果

在最终计算中，我们运用计算机实现了若干数据文本的热度分计算，使得每一文本数据都有了对应的热度指标分，并以此为最终结果排出热度分前 5 的留言事件，匹配出了其对应热度地点及热度人群，对应相应的时间梯度，实现了最终

的热度评价。具体的表格实验结果详见附件。

4. 问题 3 分析方法及过程：

4.1. 问题浅析

这一问题要求针对政府“线上问政”的答复意见评价，从“完整性”、“相关性”、“可解释性”三个方面进行量化分析，并给出一套评价方案。在对答复意见的文本数据进行总结分析后，我们拟以三个特性为标准，量化出了基本的权重比例，并将生产运作中的因素评分法[⑧]与综合评价法[⑨]有机结合，给予各个样本一定的评价标准，进行了如下的评价方案制定：

4.2. 方案评价流程

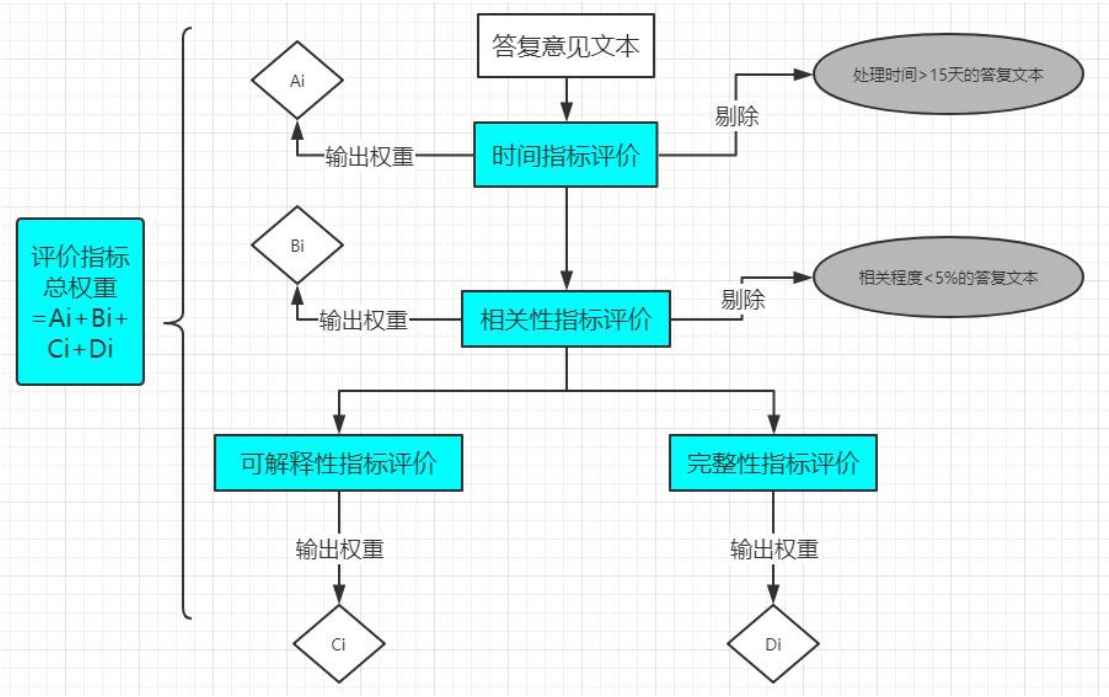

图 10 答复意见 评价方案操作流程

⑧ 因素评分法是首先从所有待评价的工作中确定几个主要因素，每个因素按标准评出一个相应的分数，然后根据待评工作总分确定相应的等级。

⑨ 综合评价法指的是运用多个指标对多个参评单位进行评价的方法，简称综合评价方法。

4.3. 指标量化

4.3.1. 相关性

(1) 相关性指标

衡量相关性，顾名思义就是衡量政府答复意见与市民线上留言反馈的问题之间的相关程度，这是答复意见评价最为重要的一环。对于线上留言问政的市民，得到的答复是否与自己提出的问题相匹配，是最重要的评价指标。因此，相关性的衡量评价，是制定本次评价方案的基础。对于相关性指标评价符合一定标准的答复文本，评价方案会继续对该文本的完整性及可解释性进行量化评价，而对于一些相关性指标评价不符合标准的答复文本，本方案会直接将该文本剔除，并给予合理的低权重评价。

(2) 文本语义网络分析

语义网络分析是指筛选统计出高频词以后，以高频词两两之间的共现关系为基础，将词与词之间的关系数值化处理，再以图形化的方式揭示词与词之间的结构关系。其基本原理是统计出文本中词汇、短语两两之间共同出现的次数，再经聚类分析，梳理出这些词之间关系的紧密程度。

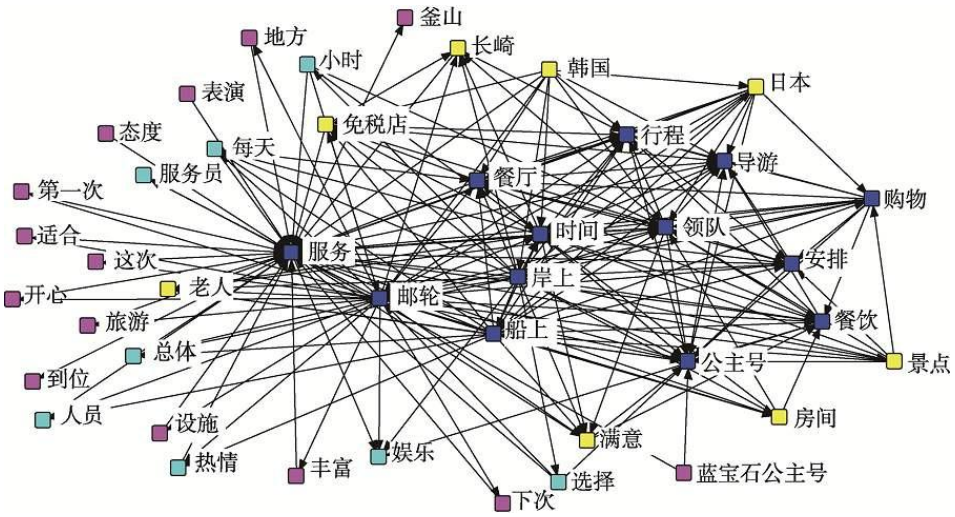

图 11 文本语义网络分析结构示意

基于相关性指标评价，我们在政府答复意见及其相应留言的文本数据中分别提取高频词，实现文本语义网络分析，生成文本语义网络结构图，直观的对高频词的层级关系、亲疏程度进行分析，由此得出留言文本与答复意见的对应关系程

度，形成相关性指标。

(3) 分析结果量化

针对文本语义网络分析的结构关系图，我们制定了六阶的相关性分析结果，将市民留言与政府答复的文本关系亲密程度划分为 A、E、I、O、U、X6 个等级。见下表：

代号	密切程度	原始占比
A	**绝对密切**	**20%**
E	**特别密切**	**15%**
I	**密切**	**10%**
O	**一般**	**5%**
U	**不密切**	**——**
X	**不靠近**	**——**

表 7 留言与答复意见文本密切程度及权重占比

由表格原始占比数据可知，作为整体评价方案的基础，整个相关性评价指标原始占比为 50%，其中，依据市民留言与政府答复内容文本的密切程度，有着不同梯度的原始占比 B_i。评价方案将以原始占比是否大于等于 5%（>=5%）为依据，即是否密切程度高于或等于“一般”，由此判断该数据是否需要继续进行完整性及可解释性指标评价的流程。原因已在 34(1)相关性指标阐述，在此不再赘述。

4.3.2. 完整性

(1) 完整性指标

完整性指标，实质上是一种格式上的指标，即判断答复意见的文本数据是否符合一定的文本规范。政府的答复意见作为一种官方的事件解读或处理，需要有官方的规范格式，以展现政府服务于民的宗旨和严肃政绩的作风。

(2) 句式提取

在对政府答复意见的文本数据进行对比分析后，我们得出了相对固定的文本句式，这些文本句式主要分布于数据结构的开头语和结语。他们之间的有机组合，构成了答复意见数据文本的完整性，而句式的具体参考内容如下所示：

开头：

亲爱/尊敬的朋友/网友,您好!

您反映的问题已了解/收到

现将具体情况/问题回复/调查结果公布如下

结尾：

感谢/谢谢您/网友的关心/支持/监督

感谢您提出的宝贵意见/建议

祝您生活愉快

如有情况将及时回复

对您造成的不便，深感抱歉

X 年（.）X 月（.）X 日（.）

联系人信息：

① 如有疑问，欢迎致电/请致电/请拨 XXXXXXXXXX（多位数字号码）

② 请咨询/联系人：XXX 工作室/院/室/部/司/所

③ 地址/联系地址

其他：

编号格式（一）（二）（三）（四）...

（1）（2）（3）（4）...

第一、第二、第三、第四...

1 2 3 4.....

一 二 三 四....

我们将答复意见的数据文本与上述提取的固定句式进行文本匹配，以最终的匹配程度来决定完整性指标的评价权重。

(3) 分析结果量化

由于完整性指标检验的只是答复意见文本的展现形式，或者说是格式规范。从留言网友的角度看，得到的答复意见形式规范是否正式，一定程度上会影响留言反馈的满意度。但并不比内容本身是否真正解决问题重要。从另一方面说，完整性是一种形式上的规范，但相比于内容本身而言，形式规范显得并没有那么重要。权衡之下，我们总共给予了该评价指标 10%的方案权重，具体评价过程如下：

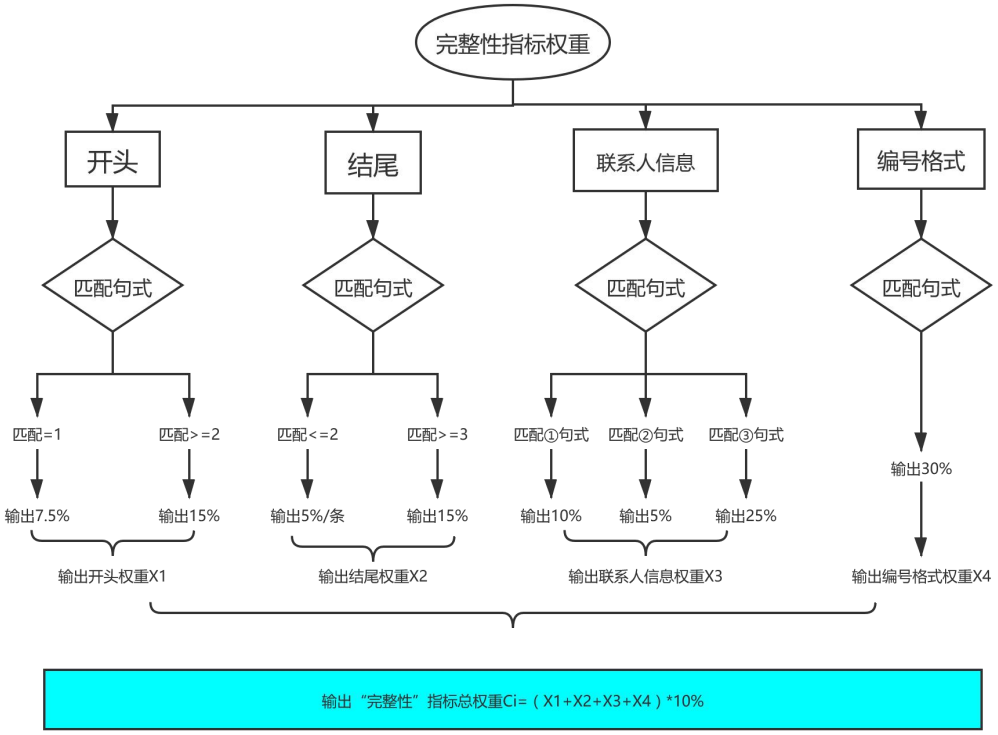

图 12 完整性评价指标 流程图

对于与“完整性”指标下提取的特定句式相匹配的答复意见数据文本，方案会给予相应分值的权重，在最终计算“完整性”指标总权重时，再乘以 10%求得其最终权重，实现评价。例：某答复数据开头匹配 2 句式、结尾匹配 1 句式、联系人信息匹配联系电话及住址 2 句式、编号格式句式不匹配，那么该数据“完整性”总权重计算过程如下：

$$C_i = (X_1 + X_2 + X_3 + X_4) \times 10\%$$

$$C_i = (7.5\% * 2 + 5\% + 10\% + 25\%) * 10\% = 5.5\% 。$$

4.3.3. 可解释性

(1) 可解释性指标

可解释性指标是用来衡量政府对市民提出的问题所进行的答复，是否真正做到了实地考察、引经据典，是否真的值得考究，具有说服力。仅仅靠口头解决的事情，是没有依据没有解决实质问题的，而可解释性指标就是为了避免假大空的问题答复。

(2) 评价依据

基于对可解释性指标的理解，我们对各个答复意见文本数据进行了对比分析，发现对于事件的解读，政府答复作为官方文件，总是包含着一定的书面依据，例如各项条例政策、政府号令、专业核实等等。由此，我们筛选出了可能进行政策引例和情况核实的标志性语句，并对此类语句与相应答复文本数据运用 python 语言进行词频，关键词匹配，以此形成了可解释性的评价指标。

符号类	引词类	编号类
《》	*号文件	一 二 三
<>	合同	1 2 3
()	通知	① ② ③
{}	依据	（1） （2） （3）

图 13 “可解释性”指标匹配词库（部分）

其中，标号格式既是完整性的评价指标，也是可解释性的一大评价标准，因为适度的编号运用，能使文本更有条理，对于解决方案的编号，摆明了事件处理的先后层次关系，能够有助于事件的说明和事务的处理，这也是可解释性的涉及范畴。同样，在进行可解释性指标处理时，需要对之前的相关性 43(3)4.4.4.4.2 相关性指标的语义网络分析结果进行再利用。语义网络分析进行了词与词之间的特定的语义连接，实现了留言和答复的重复词频统计和词汇链接，可解释性要求的是答复意见的科学性，答复意见是否科学的一大依据也正是其是否与网友留言文本切实相关。

(3) 分析结果量化

权衡之下，我们给予可解释性指标总权重占比 20%的总比例，并将上述评价指标分为三类：符号类、引词类与编号类，将 python 实现的文本匹配与语义网络分析实现的词与词之间关联的匹配频次相结合，形成下述权重方案：

符号类：由于符号类引例可信度较高，特别是“《》”“<>”等文件专用引例符，能够比较精准定位数据文本的引例次数及长短，且鉴于引例可能存在的次数局限性，故符号类引例给予了 60%的总匹配价值及 20%的权重基数。

（设：匹配次数为 k，权重基数为 20%）

引词类：引词类引例，顾名思义是以可能引出相应政策号令、相关文件的特定词汇引出相关文件，鉴于此类引词相对灵活且容易泛用，无法实现精准匹配，故引词类引例仅给予 15%的总匹配价值及 3%的权重基数。

（设：匹配次数为 n，权重基数为 3%）

编号类：编号类匹配的是文本呈现的条理性以及其影响的文本数据的可理解性，编号类指标相对固定，属于匹配机制下一对一的对应产物，考虑到“完整性”一栏已经给予不小权重，故可解释性指标中的编号类指标给予剩下 25%的总匹配价值。

（匹配次数为 d；注：编号的连续性决定其权重基数即为其总匹配价值，即当满足编号格式序列时，就给予 25%的总权重）

$$D_i = (0.2k + 0.03n + d) * 20\% (k <= 0.6, n <= 0.15, d <= 0.25)$$

（注：D_i 为可解释性指标权重，k，n，d 分别为符号类、引词类、编号类文本匹配次数）

4.3.4. 时效性

(1) 时间指标

除了相关性、完整性、可解释性之外，还有更加重要的指标，它甚至是决定一个答复意见是否有效最为关键的指标，这便是时效性。一个政府答复是否有效，最根本的是它反映事实、解决事务要及时[12]。特别是当市民进行反馈留言时，如果表示时间不够，事况紧急，而政府答复虽然细致独到，但却时效滞后，可能更会适得其反，引起市民不满。

例：

留言文本：2017/3/1

C 市 XX 小区 XX 街道已经无故停电 2 天了,有关部门也搪塞是电力输入故障，严重影响了我们的工作生活，希望政府部门能立即采取措施，没电怎么生活啊！！！

答复意见：2017/3/14

尊敬的网友，您好！据调查取证，我们了解到该地突发性停电原因在于电力工作者的认人为误操作导致的电力系统暂时性短路故障,排查后发现该地区电力储备及应急预案存在一定问题,我们已经进行了相应整改，并对该地电力供应系统进行有效排查和升级。电力是民生问题，我们一定会竭尽所能，进行有效的整治行动，还百姓一个安定合理的生活，目前问题已经得到解决，如果您还有发现其他问题，或者对我们的工作有什么意见和建议，请继续信访告知，感谢您的来信，谢谢您对政府工作的理解和支持。

以上政府答复意见条理清晰、格式完整、内容详实，对于事件原因也有进行相应核查，按理说是评价指标很高的答复文本，但对于停电这样的涉及人民生活的大问题，特别是市民已经经历了连续停电 2 天的遭遇，有关部门却在 14 天后（答复时限是 15 天）才进行答复解决，显然已经错过了最佳的解决时间，这样即使答复意见其他指标再满足，也不是一个合理的答复。

因此，我们认为时间指标不仅必要，而且是一切的根本。

(2) 文本时效确定

对于任一网友留言文本而言，都有其时效，就像中国有句古话：“凡事总归有个轻重缓急”。而对于政府答复意见来说，轻重有别地处理留言和事务，也是至关重要的。所以我们认为，对于任一答复意见的时效性指标判断，应该先以相应的留言文本为依据，分出“紧急、稍急、一般”三种等级的文本时效，并以此为基础，实现指标的量化评价。

首先，经过查询各大省份网上留言信息答复时间的硬性规定[10]，我们得出，政府基本要求，线上问政的留言答复时间为十五个日内，因此，超过十五天的答

[10] 《信访条例》第二十一条 县级以上人民政府信访工作机构收到信访事项，应当予以登记，并区分情况，在 15 日内决定是否受理并书面告知信访人，并按要求通报信访工作机构。

复文本不符合时间规定，将直接剔除出本次方案评价，为不合格答复。

其次，我们在对各留言进行数据挖掘时，提取了留言文本能够说明事项是否紧急的标志性词汇，并在合理利用 python 及 Excel 数据处理的基础上，将其爬取生成固定文本，分别标识为“紧急、稍急、一般”三类数据，实现了时效词汇分词。（注：标准如下：“紧急”留言的标准是关乎市民日常衣食住行的关键性问题；“稍急”留言的标准是涉及市民基础民生，生活质量的问题；“一般”留言的标准是为方便处理事务，形成的经常性疑问，为前两者的排除项）

紧急	稍急	一般
任何	一直	噪音
严重	就业	吵闹
高达	残疾	网速
多达	养老	保姆
安全	污染	纠纷
违法	入学	罚款
停水	违建	赔偿
断电	高考	争吵
毒	工资	地址

图 14 答复意见时效界定词库（部分）

在实现词汇分词的基础上，我们利用 Excel 工具的条件格式，并结合函数功能实现了留言文本的时效分类。分类流程为“一般”—“稍急”—“紧急”，以不放过任何一条紧急事态的线上留言为标准。就这样，我们测试完成了对于各答复意见的时效性指标，形成最终量化结果如下。

(3) 指标量化

在整体方案中，时间指标将是所有指标之中最先考察的，政府答复只有解决实事才能予以好评，而解决实事的标准，排在第一位的就是时效性。针对前面形成分类的三类不同时效的数据文本，将形成不同的梯度指标。

首先，是对数据文本的时间数据提取，每一条留言数据都有对应的留言时间

和答复时间，我们使用 Excel 的 vlookup 函数，对时间数据文本进行提取，并利用 python 语言进行文本匹配，确保每一条留言内容具有其对应的时间要素。

其次，是时间指标的总权重占比，我们总共给予权重占比 20%的指标。第一步是实现留言时间与答复时间的对比，将不符合十五天内的数据文本剔除并进行记录，然后，再对分类的三类不同时效的文本进行下述评价：

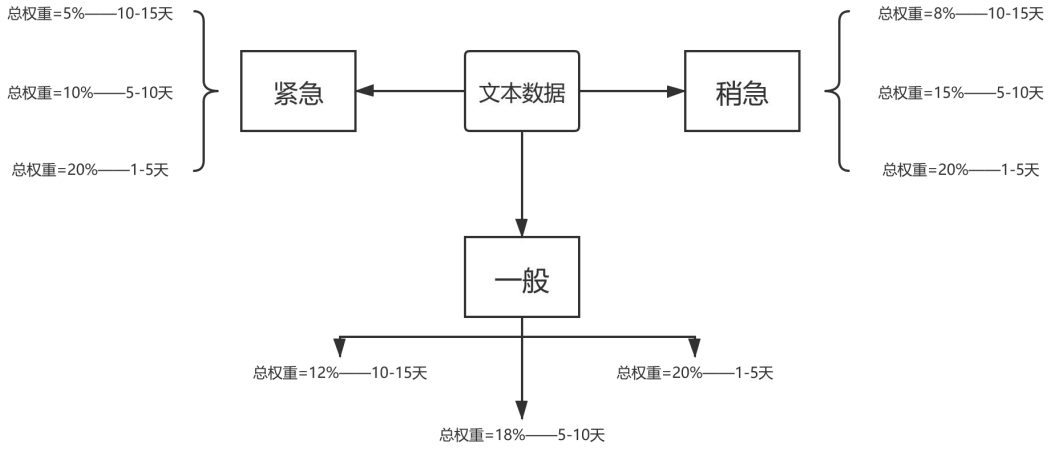

图 15 “时效性”权重指标

（注：总权重占比 A_i 不超过 20%）

在既定的三类分类标签权重基础上，回复时间越快，达到的指标占比越大；

4.4. 评价指标结果

综上，我们不难得出，相关性、完整性、可解释性指标之间的评价过程也是密不可分的，答复意见的评价，也是三个指标有机统一的结果，无法割裂来看。而对于各个特性的指标量化而言，分步骤进行评价也是必要的。因此，我们对各个指标进行了以下内容的梯度评价以最终确定答复意见效力。

4.4.1. 时间指标

首先进行的自然是时间指标的评价，时效性永远是一个事件是否得到有效处理的首要标准，因此，我们在依据时间效力是否超过相应下界为标准筛选出对时间具有不同要求程度的留言后，对其相应的政府答复意见文本进行不同梯度的权重评价 A_i（详见**错误！未定义书签。**(3)4.3.4 时效性），并剔除无用的答复数

据，进入到下一步评价操作。

4.4.2. 相关性指标

在时间评价的基础上，进行的是相关性的评价，以文本语义网络分析结果为依据，实现了留言与答复之间不同相关程度的分类（详见 434.4.2 相关性指标），给予不同权重的评判 B_i，并再次剔除与留言基本无关的答复数据。

4.4.3. 可解释性与完整性指标

而后进行的分别是可解释性和完整性评价，两者课同步进行，分别对政府答复意见的科学性以及语言格式的规范性进行权重评判，得出权重占比 C_i 与 D_i（详见 434.4.3 可解释性与完整性指标）。

4.4.4. 最终评价指标

最终，所有答复意见数据文本（剔除项除外）都实现了四个指标的量化评价，其各自总评即为

$$Z_i = A_i + B_i + C_i + D_i$$

（注：Z_i 为答复意见总权重，且上述皆为百分比数值，且不超过其指标所占的评价方案总权重）

公式 12 评价方案总权重计算

5. 结论

本文通过对政务留言及政府答复意见的文本留言数据利用 python 编程语言进行基于三种主要算法（XGBoost、GDBT、SVM）的数据挖掘模型建立，得到了具有一定价值的结果，实现了对文本留言数据的一级分类以及基于对用户留言文本、互动行为，时间梯度等指标的热度事件挖掘，这些结果对于政府有关部门处理网上政务留言以及关注民生热点等工作都具有一定的指导意义。比如依照热度

评分指标，政府相关工作人员能够借助计算机了解到市民关注的热点事件，并做好相关调查处理工作，保证事件的跟进和问题的解决，能够更好地服务于人民，深入群众，做好“人民的政府”，提高市民对政府处理网上政务问题的满意度以及对政府的信任度。

但从另一方面来看，从我们的文本数据挖掘结果来看，也可以看出我们的数据处理过程并不是一帆风顺或完美无缺的。我们在文本数据处理中，尝试了多个模型及算法进行文本数据的分类及处理，得到的结果虽然较为可观，但缺陷在于文本数据处理量的庞大以及存在的一些较为主观性质的热度判断。这里面涉及到的包括有中文语言结构的复杂表达，还有就是当今对于中文文本数据挖掘模型的不足以及网民留言数据本身所具有的问题等等，这些也将是我们在往后对中文文本数据研究过程中应该持续思考、深入探讨、继续改进的地方。

6. 附录

6.1. XGBoost 算法

```
#模型一XGBoost
import xgboost as xgb
#xgboost.train()利用param列表设置模型参数
#XGBoost训练参数列表
#减半步长，最大迭代次数加倍来增加我们模型的拟合能力和泛化能力
params = {
    'booster': 'gbtree',          #采用树的结构运行数据，gblinear基于线性模型
    'objective': 'multi:softmax', #目标函数使用Xgboost做多分类问题，需要设置num_class
    'num_class': 7,               #本题分类的个数是7个
    'gamma': 0.2,                  # 用于控制是否后剪枝的参数,只有损失函数下降超过这个值节点才会越大越保守，一般0.1、0.2这样子。
    'max_depth': 6,               # 构建树的最大深度，值越大越复杂（3->5->6->7->6->8->7->6）
    'lambda': 4.,                   # 控制模型复杂度的权重值的L2正则化项参数，参数越大，模型越不容易过拟合。4.->5.->4.
    'subsample': 0.4,              # 每棵树随机采样30%的训练样本，通常取0.5-1，（0.3->0.5->0.3->0.4）
    'colsample_bytree': 0.5,       # 生成树时进行的列采样率，特征采样率
    'min_child_weight': 1.2,       #最小孩子节点，一个子集所有观察值的最小权重和
    'silent': 0,                   # 设置成1则没有运行信息输出，最好是设置为0.
    'eta': 0.007,                   #更新中收缩步长，在每次提升计算之后，算法会直接获得新特征的权重。 eta通过缩减特征的权重使提升计算过程更加保守
    'seed': 1000,                  #随机数种子
    'nthread': 4,                 #更新线程数，缺省值是当前可获得的最大线程数
}
#训练booster模型，使用xgboost的内部数据结构
dtrain = xgb.DMatrix(trainx, Trainy)
dtest = xgb.DMatrix(testx)
num_rounds = 500#迭代次数500->600->500
model = xgb.train(params, dtrain, num_rounds)
#引入F1-Score构建评估函数
from sklearn.metrics import f1_score
from sklearn.metrics import classification_report
# 训练集，获取训练集预测结果
train = xgb.DMatrix(trainx)
predy1 = model.predict(train)
# 测试集，获取测试集预测结果
predy2 = model.predict(dtest)
```

6.2. 梯度提升树 GDBT

```
#模型二梯度提升树GDBT
from sklearn.ensemble import GradientBoostingClassifier

gbdt = GradientBoostingClassifier(loss='deviance', learning_rate=0.01,#使用默认对数似然损失函数，对多元分离有比较好的优化，学习率0.01->0.001
                                  n_estimators=50, subsample=0.3 #和XGB一样的百分之3十随机子采样
                                  , min_samples_split=2, min_samples_leaf=1, #内部节点再划分最小样本数为2，叶子节点最少样本数为1
                                  max_depth=4, init=None, random_state=None,#决策树最大深度3->6->4
                                  max_features=None,verbose=0,#划分时默认考虑所有的特征数
                                  max_leaf_nodes=None, warm_start=False #不限制最大叶子节点数，不添加新的决策树
                                  )
gbdt.fit(trainx, Trainy)


# 训练集
predy1 = gbdt.predict(trainx)
# 测试集
predy2 = gbdt.predict(testx)
```

6.3. SVM 支持向量机

```
#模型三SVM支持向量机
from sklearn.svm import SVC
from sklearn.multiclass import OneVsRestClassifier

classif = OneVsRestClassifier(SVC(kernel='linear',   #错误项的惩罚系数为1.2，采用线性核参数
                                  probability=True        #启用概率估计
                                  ))
classif.fit(trainx, Trainy)

print("已完成SVM模型的训练和测试，结果如下")
predy1 = classif.predict(trainx)
predy2 = classif.predict(testx)
```

7. 参考文献

[1] 众里塔.缺失值处理.[J / OL].CSDN-专业 IT 技术社区.2016-12-27

[2] 达观数据 DataGrand.中文文本分类：你需要了解的 10 项关键内容.[J/OL].2018-10-26

[3] 陈莉萍，杜军平. 突发事件热点话题识别系统及关键问题研究[J]. 计算机工程与应用，2011，47(32).

[4] 马治涛. 文本分类停用词处理和特征选择技术研究[D].西安电子科技大学,2014.

[5] 西多士 NLP.词向量(one-hot/SVD/NNLM/Word2Vec/GloVe).[J/OL].博客园.2019-09-29

[6] 陈运文.一文详解 LDA 主题模型.[J/OL].知乎社区.2018.12.05

[7] v_JULY_v .支持向量机通俗导论（理解 SVM 的三层境界）.[J/OL].CSDN-专业 IT 技术社区.2012-06-01

[8] yip522364642.基于支持向量机 SVM 的文本分类的实现.[J/OL].CSDN-专业 IT 技术社区. 2017-01-16

[9] 泽翾 数据清洗.[J/OL].CSDN-专业 IT 技术社区.2019-02-27

[10] 人人都是产品经理.产品经理需要了解的算法：热度算法和个性化推荐.[DB/OL].CSDN-专业 IT 技术社区.2018.6.7
[11] 王晓光,王宏宇,黄菡.基于多源数据的专业领域热点探测模型研究[J].图书情报工作,2019,63(14):52-61.
[12] 姚慧玮. 关于网上信访运行机制的研究[D].华东政法大学,2014.