摘要

随着国家政策的发布，互联网问政平台的构建，网络也成为政府了解民意的重要途径。各级政府在处理电子政务方面有了迫切的需求，智慧政务应运而生。借助大数据、云计算、人工智能，基于自然语言处理技术的智慧政务系统能够解决以往主要依靠人工处理的留言划分和热点整理等工作，大大提升了政府的管理水平和效率。

对于本次赛题中给出的来自互联网公开来源的群众问政留言记录以及相关部门对部分群众留言的答复意见，我们运用自然语言处理和文本挖掘的方法解决在处理过程中产生的问题。

对于处理群众留言分类，我们先将数据集进行预处理后，使用 jieba 分词将中文文本切割。再对处理之后的文本通过改进的 TF-IDF 算法提取文本特征词，得出结果后使用朴素贝叶斯预测种类，并根据属性之间的依赖关系添加隐藏的父节点，增强了属性之间的依赖关系，提高了情感分类的准确性。我们的模型正确率为 92.27%, 错误率仅为 7.73%，最后我们用 python 引用 Scikit-learn 中的 F-Score 模块对分类方法做出评价, 得出 $F1-Score=0.9226$。

对于热点问题挖掘，我们在对数据集预处理后，使用 gensim 分词形成的二维数组生成字典，将二维数组通过 doc2bow 稀疏向量形成语料库。再用 LSI Model 算法，将语料库计算出 TF-IDF 值，并获取词典 token2id 的特征数，计算出稀疏矩阵相似度，建立一个索引，最后通过 doc2bow 计算测试数据的稀疏向量以求得测试数据与样本数据的数量。最后我们根据反应同种问题的数量和时间范围的长短，定义了热度评价指标。然后得出 A 市伊景园滨河苑捆绑销售车位等 5 个热点问题。

对于答复意见的评价，我们首先数据清洗，然后对每条“留言详情”和“答复意见”文本分词，去停用词。我们构建了专业的关键词词典，利用 word2vec 算法对智慧政务系统的历史数据库进行训练，得到该数据库的词向量和对应词语的相似度。从相关性、完整性、可解释性、内容质量、时效性 5 个一级指标和信息贴进度等 8 个二级指标构建相关部门对留言的答复意见的评价模型。

关键词： 智慧政务　TF-IDF　朴素贝叶斯　Dord2vec　LSI Model

目录

一、引言 1

1.1 问题背景 1

1.2 问题重述 1

1.3 问题分析 2

二、数据预处理 2

2.1 中文分词 2

2.2 基于 jieba 的中文文本切割 3

2.3 分词结果示例 4

2.4 词频统计 6

三、基于 TF-IDF 的朴素贝叶斯文本分类模型 8

3.1 TF-IDF 算法 8

3.2 朴素贝叶斯算法原理 10

3.3 模型的求解与结论 11

3.3.1 去掉停用词 11

3.3.2 实验结果 11

3.4 评价指标 14

四、文本相似度热度统计 14

4.1 模型的基础 15

4.1.1 Gensim 15

4.1.2 潜在语义索引 (LSI) 15

4.2 Word2vec 基本原理 17

4.3 Doc2vec 基本原理 18

4.4 模型算法的实现 19

4.5 热度评价指标 19

4.6 模型的实验结果 19

五、留言回复质量评价模型 20

5.1 基于 TextRank 的留言回复质量的关键词提取 20

5.2 留言回复质量指标体系的确立 22
5.3 模型构成 22
5.4 基于 AHP 的指标权重计算 23
5.4.1 基于指标归类的两两判断矩阵 23
5.4.2 权重向量及一致性检验 23
5.4.3 指标权重的计算 24
5.5 子模型介绍 24
5.5.1 相关性评分模型 24
5.5.2 完整性评分模型 25
5.5.3 可解释性评分模型 25
5.5.4 回复内容质量模型 26
5.5.5 及时性模型 26
5.6 模型的实现 26
5.6.1 建立评价等级 27
5.6.2 实验结果 27
六、总结与后续研究建议 28
6.1 全文总结 28
6.2 后续研究建议 29
参考文献 30

一、 引言

1.1 问题背景

近年来，各级政府强调在政务工作中要做到广泛征求民众意见，扩大公众的政治参与，保障公民积极行使自身权利，已经成为了各级政府了解民意、汇聚民智、凝聚民气的重要渠道。各级政府广泛开放了微信、微博、市长信箱、阳光热线等网络问政平台，公众开始在各类平台上积极参与政务讨论，抒发自己的意见，同时也开始以各种渠道和方式，参与到政治事务当中。与此相对应的，是数量剧增的各类留言和信件，各类与社情民意相关的文本数据不断增多，给政务信息处理带来了极大的挑战。而目前，计算机技术正飞速发展，从社会生活的各方面深度影响着人类生活的方方面面，其中也包括了政府利用计算机技术开展公共管理服务，利用自然语言处理技术的智慧政务系统已经是社会创新治理的新方向。信息化和全球化的趋势正在将政务处理带入一个崭新的信息化时代，同时紧随而来的还有信息爆炸的问题，如何快速地从海量的公众信息文本中提取到关键有用的信息已经成为当今智慧政务系统关注的热点。

1.2 问题重述

文本挖掘近年来已经成为一门相对独立的研究学科。最早可追溯到二十世纪九十年代初期，美国所提出的国家信息基础设施计划，其极大地推动了全球的信息化进程，十多年后，包括我国在内，人们可以便捷地获取和传输各种信息，使得全球各种渠道内的信息量呈指数式增长，其中以互联网信息文本的数量增长为主要代表。根据查阅整理分析文献了解到，我们所获取并存储的信息，绝大部分是以文本的形式，保存在文本文档中，文本也是互联网应用中最为常见的储存形式，其中包括储存在各种新闻媒体、研究论文、书籍平台、电子邮箱和网页等地方的各类信息文本。美国国家标准与技术研究所和美国国防高级研究计划局组织的文本检索会议是国际上著名的文本挖掘评测会议。始于 1992 年，以每年一届的频次开展，主要关注评测文档检测、信息提取和摘要三项基础文本技术。文本挖掘技术是从大量的文本数据中去发掘隐藏的规律，从而摘取重要信息的过程。

“智慧政务”即是政府机构利用现代云计算、人工智能、大数据等技术，将政府管理和服务功能通过互联网进行实现，在互联网上实现了政府组织机构的工作流程，这种方式避免了部门之间时空间隔，有效提高了政府机构的工作效率和透明度。现如今“智慧政务”电子系统正在快速发展，各级政府收集整理的各种民意文本数据量快速增长，形成了“数据爆炸”的局面，而政府处理解决这些意见的方式通常还是以人工进行分析和处理，其分析处理文本数据的能力并没跟上信息增长的速率。政府此时正需要一种高效的文本挖掘技术，处理其从各种平台收到的信息反馈，高效准确地提取有效信息，提高政府决策的科学性和规范性，从而更有利于建立规范的评价体系，使政务能力的提升

形成一种良性循环。

1.3 问题分析

“智慧政务”中的文本挖掘应用主要致力于从三个方面对数量众多的民众信息进行处理，其也正是智慧政务系统目前所面临数据压力所亟需解决的问题。

- 第一，对群众留言信息进行分类。工作人员需按照分类三级标签体系对政务平台的群众留言进行分类，根据相对应的部门职能，将该类信息分派至工人员手中。将分类流程通过文本挖掘技术进行操作，将极大提高工作效率，以及降低工作差错。
- 第二，挖掘群众反映的热点问题。通过文本挖掘技术的应用将群众所反映的特定地点特定人群的热点问题按照一定标准进行评估和整理，及时整理出热点问题，有助于相关人员高效做出应对方案。
- 第三，制作答复意见的评价体系。相关部门对于群众留言和来信的答复由于质量参差不齐，需要对其答复效果进行评估，从而提升政府机关解决相关问题的质量。文本挖掘技术的应用可以从政府相关部门的答复意见中，根据相关性、完整性、可解释性、合法性等角度进行筛选和评估，形成一套完整的评价体系，用于反馈政务工作的实际效果。

二、 数据预处理

数据预处理是指在主要的处理以前对数据进行的一些处理。对于题目中给定的数据集，包括群众留言、热点问题。对于这些原始数据，我们要做一些处理，将原始数据转换为预测模型易于处理的数据类型，出去噪声，遍历的读取一个文件下的每个文本。

2.1 中文分词

中文分词就是将中文连续的字序列按照一定的规则重新组合成词序列的过程，是中文信息处理的基础。[9] 自然语言处理的过程中词语往往表述为具有独立语法、语义的最小的自然语言构成单位，要进行自然语言的处理，必然需要选择一个单位来对语言进行分割处理研究。在西文中，词与词之间有明显的标记，由空格隔开，词语不需要进行其它处理；而在汉语、日语等东方语言中，由于词与词之间没有明显的切分标记，要对文本进行研究则需要将文本进行切分，即分词。此外，还需要对词语进行词性标记等。

中文文本的自动分词技术是智能检索、机器翻译、文献标引、自然语言理解与处理的基础，也是中文文本分类的一个关键的环节，对中文进行处理首先要对文本进行分词。中文分词的理论与方法决定了中文分词系统的性能与效率。同比于西文文本分类相比较，中文文本分类不同于西文文本分类的一个重要环节就是文本数据的预处理，对中文文本进行分词处理时文本预处理的首要环节。目前，中文分词技术的发展阶段从最初

的基于词典的方法，发展到了基于统计语言模型的分词方法并且已经取得了较好的研究效果。国内主流的对分词系统所做的研究中，正在研究的或者采用的方法主要有三种：基于词典规则的机械分词方法、基于人工智能的分词方法以及基于语料库的统计分词方法。

1. 基于词典规则的机械分词方法: 它是按照一定的策略将待分析的汉字串与一个“充分大的”机器词典中的词条进行匹配。若在词典中找到某个字符串，则匹配成功 (识别出一个词)。该方法有三个要素，即分词词典、文本扫描顺序和匹配原则。文本的扫描顺序有正向扫描、逆向扫描和双向扫描。匹配原则主要有最大匹配、最小匹配、逐词匹配和最佳匹配。
2. 基于人工智能的分词方法: 其基本思想就是在分词的同时进行句法、语义分析，利用句法信息和语义信息来处理歧义现象。它通常包括三个部分：分词子系统、句法语义子系统和总控部分。在总控部分的协调下，分词子系统可以获得有关词、句子等的句法和语义信息来对分词歧义进行判断，即它模拟了人对句子的理解过程。这种分词方法需要使用大量的语言知识和信息。目前基于理解的分词方法主要有专家系统分词法和神经网络分词法等。由于汉语语言知识的笼统、复杂性，难以将各种语言信息组织成机器可直接读取的形式，因此目前基于理解的分词系统还处在试验阶段。
3. 基于语料库的统计分词方法: 该方法的主要思想：词是稳定的组合，因此在上下文中，相邻的字同时出现的次数越多，就越有可能构成一个词。因此字与字相邻出现的概率或频率能较好反映成词的可信度。可以对训练文本中相邻出现的各个字的组合的频度进行统计，计算它们之间的互现信息。互现信息体现了汉字之间结合关系的紧密程度。当紧密程度高于某一个阈值时，便可以认为此字组可能构成了一个词。该方法又称为无字典分词。该方法所应用的主要的统计模型有：N 元文法模型、隐 Markov 模型和最大熵模型等。在实际应用中一般是将其与基于词典的分词方法结合起来，既发挥匹配分词切分速度快、效率高的特点，又利用了无词典分词结合上下文识别生词、自动消除歧义的优点。

2.2 基于 jieba 的中文文本切割

我们本次分词用 jieba 分词来进行辅助，下面我们用一幅框架图来说明 jiaba 分词(如图1所示)：

jieba 分词有以下几个类型：

1. **精确模式**: 试图将句子最精确地切开，适合文本分析；
2. **全模式**: 把句子中所有的可以成词的词语都扫描出来，速度非常快，但是不能解决歧义。

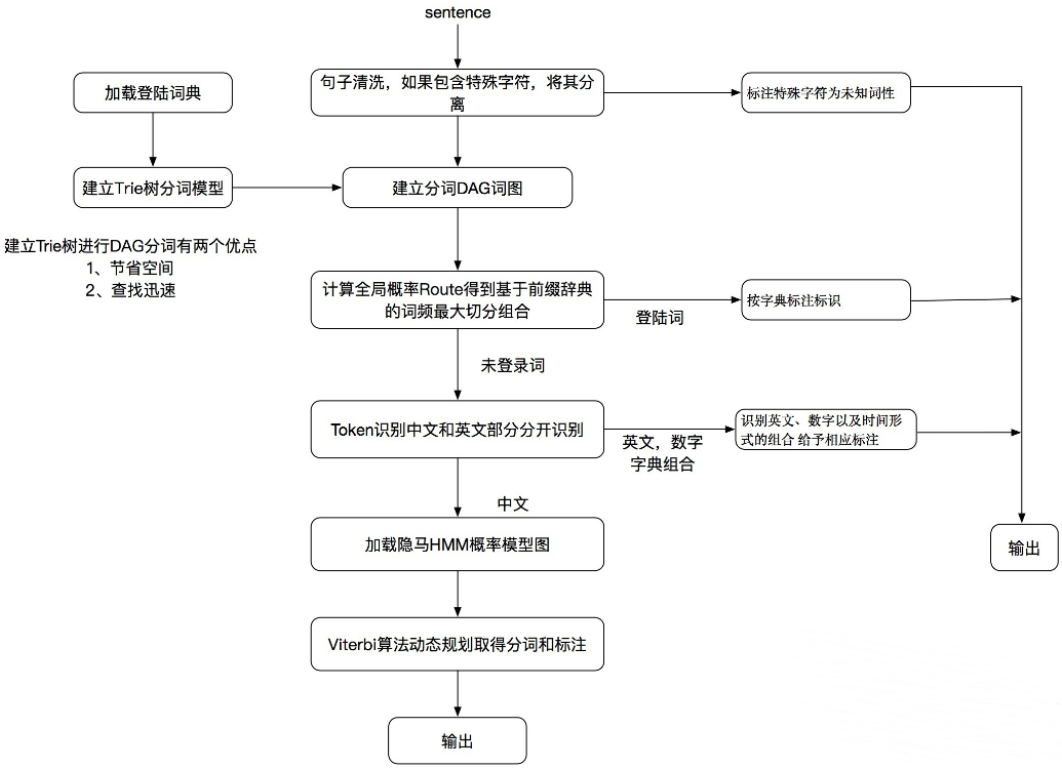

图 **1** **jiaba** 分词的框架图

3. **搜索引擎模式**: 再将却模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词

以下举了几个例子：

- **精确模式**: A/ 市/ 西湖/ 建筑/ 集团/ 占道/ 施工/ 有/ 安全/ 安全隐患/ 隐患
- **全模式**:A/ 市/ 在水一方/ 大厦/ 人为/ 烂尾/ 多年/ ，/ 安全隐患/ 严重
- **搜索引擎模式**:A/市/魅力/魅力之城/商铺/无/排烟/管道/排烟管道/小区/区内/小区内/到处/油烟/烟味/油烟味

因为题目所给的是文本数据，所以我们使用的是精准模式，便于进行文本分析。

2.3 分词结果示例

我们对赛题官网给出的文本数据进行了分词, 经过分词处理之后，原始的文本会被处理成一系列由 | 隔开的单个词。表(1)将展示部分数据的分词结果

表 **1**　分词结果部分数据

分词前	分词后
K8 县丁字街的商户乱摆摊	K8| 县 | 丁字街 | 的 | 商户 | 乱 | 摆摊
K8 县九亿广场的公厕要安装照明灯	K8| 县 | 九亿 | 广场 | 的 | 公厕 | 要 | 安装 | 照明灯
K 市域轨道交通规划建议	K| 市域 | 轨道交通 | 规划 | 建议
关于 K 市域轨道交通规划的建议	关于 |K| 市域 | 轨道交通 | 规划 | 的 | 建议
咨询 A 市楼盘集中供暖一事	咨询 |A| 市 | 楼盘 | 集中 | 供暖 | 一事
A 市可以实现集中供暖吗？	A | 市 | 可以 | 实现 | 集中 | 供暖 | 吗 | ？
K9 县坐公交车要 2 元？	K9 | 县 | 坐 | 公交车 | 要 | 2 | 元 |？
反映 K6 县公交车监控的有关问题	反映 |K6| 县 | 公交车 | 监控 | 的 | 有关 | 问题
请把 L 市公园里的门球场尽快修好	请 | 把 |L| 市 | 公园 | 里 | 的 | 门 | 球场 | 尽快 | 修好
K1 区珠山街的下水设施需要完善	K1 | 区 | 珠山 | 街 | 的 | 下水 | 设施 | 需要 | 完善
投诉 A 市 A1 区苑物业违规收停车费	投诉 |A | 市 | A1| 区苑 | 物业 | 违规 | 收 | 停车费
投诉 A 市盛世耀凯小区物业无故停水	投诉 | A | 市 | 盛世 | 耀凯 | 小区 | 物业 | 无故 | 停水
A 市泉塘的道路绿化问题需要重视	A| 市泉塘 | 的 | 道路 | 绿化 | 问题 | 需要 | 重视
希望 L9 县好好保护现存的清代建筑	希望 |L9| 县 | 好好 | 保护 | 现存 | 的 | 清代 | 建筑
强烈要求 M4 市湾田学校不要盲目改址	强烈要求 |M4| 市湾田 | 学校 | 不要 | 盲目 | 改址
投诉 I6 市新时空小区不发放房产证	投诉 |I6| 市 | 新时空 | 小区 | 不 | 发放 | 房产证
A 市联美地产建最牛渗水楼盘	A| 市 | 联美 | 地产 | 建 | 最 | 牛 | 渗水 | 楼盘
A 市楚华宾馆后棚户改造建议	A| 市楚华 | 宾馆 | 后 | 棚户 | 改造 | 建议
K5 县潇水壹号工程施工违规招标	K5| 县 | 潇水 | 壹号 | 工程施工 | 违规 | 招标
D10 县联兴神威拖欠工程款	D10| 县联兴 | 神威 | 拖欠 | 工程款
关于申请 C 市公租房问题的咨询	关于 | 申请 | C | 市公 | 租房 | 问题 | 的 | 咨询
C4 市 2019 年廉租房什么时候分下来	C4| 市 |2019| 年 | 廉租房 | 什么 | 时候 | 分 | 下来
K3 县龙腾大厦问题好多	K3| 县龙腾 | 大厦 | 问题 | 好多

2.4 词频统计

为了能精确的对实验文档进行表示，基于统计的方法需要对整篇文章进行分词，进而对每个单词的出现频率进行统计。词是最小的、能独立活动的、有意义的语言成分。计算机的所有语言知识都来自机器词典 (给出词的各项信息)、句法规则 (以词类的各种组合方式来描述词的聚合现象) 以及有关词和句子的语义、语境知识库。

在中文文档中，从形式上看，词是组合稳定的字而得到的，因此，在上下文环境中，同时出现相邻的字次数越多，就表示该相邻字组合越有可能构成一个词。因此相邻字的组合共现的频率或概率能够较好的反应组成词的可信度。这就是词频统计的基本原理，这种技术发展至今已经有许多不同的统计原理。

1. **互信息原理**: 对有序汉字串 AB 中汉字 AB 之间的互信息定义如公式：

$$log = \frac{P(A,B)}{P(A)P(B)} \tag{1}$$

互信息体现了汉字之间结合关系的紧密程度当紧密程度高于某一个阈值时便可认为该字组合可能构成了一个词。其中,P(A,B) 为汉字串 AB 联合出现的概率,P(A) 为出现汉字串 A 的概率，P(B) 为汉字串 B 出现的概率。

2. **N-Gram 统计模型**:N-Gram 统计计算语言模型的思想是一个单词的出现与其上下文环境中出现的单词序列密切相关, 第 n 个词的出现只与前面 n-1 个词相关，而与其它任何词都不相关设 $W_1, W_2, \cdots, W_n$ 是长度为 n 的字串，则字串 W 的似然度用方程表示如公式：

$$P(W) = \prod_{i=1}^{n} P(W_i|W_{i-n+1}W_{i-n+2}\dots W_{i-1}) \tag{2}$$

不难看出，为了预测词 W_n 的出现概率，必须知道它前面所有词的出现概率。从计算上来看，这种方法太复杂了。如果任意一个词 W_i 的出现概率只同它前面的两个词有关，问题就可以得到极大的简化。这时的语言模型叫作三元模型 (tri-gram) 如公式：

$$P(W) \approx P(W_1)P(W_2|W_1) \prod_{i=3\dots n} P(W_i|W_{i-2}W_{i-1}) \tag{3}$$

符号 $\prod_{i=3,\cdots,n} P(W_i|W_{i-2}W_{i-1})$ 示概率的连乘。一般来说，N 元模型就是假设当前词的出现概率只同它前面的 N-1 个词有关。重要的是这些概率参数都是可以通过大规模语料库来计算的，比如三元概率如公式：

$$P(W_i|W_{i-2}W_{i-1}) \approx \frac{count(W_{i-2}W_{i-1}W_i)}{count(W_{i-2}W_{i-1})} \tag{4}$$

式中 count(⋯) 表示一个特定词序列在整个语料库中出现的累计次数。

3. **基于匹配的词频统计方法**: 单关键词匹配算法在国内外都对其进行了深入的研究，主要有 BF(Brute Force) 算法、KMP (Knuth-Morris-Pratt) 算法、BM(Boyer-Moore) 算法等，都已基本上完善。BF 算法直观、简单，但涉及多次回溯，算法效率低，时间复杂度为 O(m*n)；KMP 算法实现了无回溯匹配，避免了 BF 算法中频繁的回溯，文本串中的每个字符只匹配一次，普遍提高了模式匹配的工作效率，时间复杂度为 O(n+m)；BM 算法实现了跳跃式匹配，文本串中的部分字符不需要匹配,时间复杂度为 O(m*n)，但其最优情况下的时间复杂度为 O(n/m)。

本文对留言信息分别进行词频统计，计算出重复的关键词出现次数，然后直接按照由大到小进行词频统计排序。部分统计结果如下：

表 **2**　词频统计排序

词语	频数	词语	频数	词语	频数	词语	频数	词语	频数
我们	14384	西地省	2635	文件	1472	教育局	1434	知道	1350
没有	7378	业主	2632	办理	1469	开发商	1369	多次	1334
领导	5030	学生	2626	企业	2024	建设	1772	因为	1300
一个	4706	规定	2538	就是	1997	作为	1766	家长	1269
公司	4018	国家	2488	这个	1942	本人	1726	人民	1264
学校	4009	不能	2376	生活	1938	一直	1700	劳动	1260
问题	3985	情况	2366	已经	1937	自己	1695	社会	1249
工作	3669	希望	2347	工资	1933	这些	1691	项目	1246
现在	3567	进行	2343	政策	1911	尊敬	1685	社保	1246
政府	3343	要求	2301	教育	1911	时间	1658	村民	1233
部门	3212	单位	2274	什么	1841	老百姓	1627	反映	1224
可以	2952	人员	2272	为什么	1814	不是	1594	造成	1197
相关	2916	这样	2201	职工	1806	但是	1565	任何	1187
他们	2763	解决	2171	有关	1802	管理	1532	关于	1183
严重	2667	居民	2110	孩子	1801	发展	1529	为了	1180
小区	2638	教师	2096	老师	1782	退休	1517	房屋	1179

为方便观察词频统计中的信息特征，我们把词频较低的词语和无关词语剔除掉，将词频转换为词云图如下，词的大小代表该词在文本中出现的频率，词越大，表示出现的频率越高。

图 **2**　词频统计图

三、 基于 TF-IDF 的朴素贝叶斯文本分类模型

目前对以朴素贝叶斯算法为代表的文本分类算法，普遍存在特征权重一致，考虑指标单一等问题。为了解决这个问题，提出了一种基于 TF-IDF 的朴素贝叶斯改进算法。该算法以 TF-IDF 为基础，对处理之后的文本开始用 TF-IDF 算法进行单词权值的计算。再去掉停用词后，使用朴素贝叶斯进行分类预测。因此该算法能较好地提高分类性能，并且对不易区分的类别也能在一定程度上达到良好的分类效果。

3.1 TF-IDF 算法

TF-IDF(term frequency－inverse document frequency，词频-逆向文件频率) 是一种用于信息检索 (information retrieval) 与文本挖掘 (text mining) 的常用加权技术。TF-IDF 是

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。TF-IDF 的主要思想是：如果某个单词在一篇文章中出现的频率 TF 高，并且在其他文章中很少出现，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。

1. TF 是词频 (Term Frequency), 词频 (TF) 表示词条 (关键字) 在文本中出现的频率。这个数字通常会被归一化 (一般是词频除以文章总词数), 以防止它偏向长的文件。
   公式：

$$tf_{ij} = \frac{n_{i,j}}{\sum_k n_{i,j}} \tag{5}$$

   其中 n_{ij} 是该词在文件 d_j 中出现的次数，分母则是文件 d_j 中所有词汇出现的次数总和。
   即：

$$TG_\omega = \frac{\text{在某一类中词条}\omega\text{出现的次数}}{\text{该类中所有的词条数目}} \tag{6}$$

2. IDF 是逆向文件频率 (Inverse Document Frequency), 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。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。
   公式：

$$idf_i = \log \frac{|D|}{|(j : t_i \in d_j)|} \tag{7}$$

   其中，$|D|$ 是语料库中的文件总数。$|(j : t_i \in d_j)|$ 表示包含词语 t_i 的文件数目 (即 $n_i, j \neq 0$ 的文件数目) 但在计算过程中，会出现某一词并未在某一文本中出现的情况。为了防止出现这种分母为零的现象，最常用的方法是使用拉普拉斯平滑对上述公式进行处理，因此一般情况下使用 $1 + |(j : t_i \in d_j)|$
   进行平滑处理后的公式为：

$$IDF = \log(\frac{\text{语料库的文档总数}}{\text{包含词条}\omega\text{的文档数} + 1}) \tag{8}$$

3. TF-IDF 实际上是：TF*IDF. 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。因此，TF-IDF 倾向于过滤掉常见的词语，保留重要的词语。
   公式：

$$TF - IDF = TF * IDF \tag{9}$$

即:

$$\omega_{ij} = tf_{ij} * \log(\frac{\text{语料库的文档总数}}{\text{包含词条}\omega\text{的文档数}+1}) \tag{10}$$

3.2 朴素贝叶斯算法原理

朴素贝叶斯，简单来说就是对于给出的待分类项，求解在此项出现的条件下各个类别出现的概率的最大值。[1] 那么就可以将其归类于所求解出的最大值所在项所属的类别。朴素贝叶斯采用一个重要的假设：每个特征之间相对独立的“朴素”假设。例如“名字”中的“名”和“字”出现的概率与这两个字相邻组成词没有任何关系。基于朴素贝叶斯分类器的文本分类算法, 是目前文本文档分类算法中最有效的方法之一。贝叶斯方法的分类目标是, 在给定描述文本的特征值 $<\omega_1, \omega_2, \cdots, \omega_n>$, 得到最可能目标值 ν_{MAP}。

$$\nu_{MAP} = \arg max P(\omega_1, \omega_2, \cdots, \omega_n) \tag{11}$$

使用贝叶斯公式将此表达式重写为：

$$\nu_{MAP} = \arg max \frac{P(\omega_1, \omega_2, \cdots, \omega_n | C_j)}{P(\omega_1, \omega_2, \cdots, \omega_n)} \tag{12}$$

朴素贝叶斯分类器基于一个简单的假定: 在给定目标值时属性之间相互条件独立。换言之, 该假定说明在给定实例的目标值情况下, 观察到联合的 $w_1, w_2, \cdots, w_n$ 的概率等于单个属性的概率乘积:

$$P(w_1, w_2, \cdots, w_n | C_j) = \prod_i P(w_i | C_j) \tag{13}$$

将其带入到上式, 可得到朴素贝叶斯分类器所使用的方法。

$$\nu_N B = \arg max P C_j \prod_i P(w_i | C_j) \tag{14}$$

其中,$\nu_N B$ 表示朴素贝叶斯分类器输出的目标值。$\nu_N B$ 就是在候选分类中找出哪个类别最容易出现该文中的单词。在实际的分类过程中, 为了避免出现 $P(w_i|C_j) = 0$ 的情况, 对 $P(w_i|C_j)$ 采用下式进行估计:

$$\frac{n_k + 1}{n + |Vocabulary|} \tag{15}$$

其中 n 为该类别中出现的特征的总数,n_k 代表特征 w_i 出现的次数。|Vocabulary| 为训练集的特征的总数。

3.3 模型的求解与结论

3.3.1 去掉停用词

表 3　停用词部分数据

啊	把	别的	不然	除了	等	反之	故	哩
阿	罢了	别说	不如	此	等等	非但	故此	连
哎	被	并	不特	此间	地	非徒	固然	连同下
哎呀	本	并且	不惟	此外	第	否则	关于	两者
哎哟	本着	不比	不问	从	叮咚	嘎	管	了
唉	比	不成	不只	从而	对	嘎登	归	临
俺	比方	不单	朝	打	对于	该	果然	另
俺们	比如	不但	朝着	待	多	赶	果真	另外
按	鄙人	不独	趁	但	多少	个	过	另一方面
按照	彼	不管	趁着	但是	而	各	哈	论
吧	彼此	不光	乘	当	而况	各个	哈哈	的
吧哒	边	不过	冲	当着	而且	各位	呵	吗
呼哧	别	不仅	除	到	而是	各种	和	慢说呢
乎	即使	不拘	除此之外	得	而外	各自	何	漫说
哗	几	不论	除非	的	而言	给	何处	冒
还是	几时	不怕	自家	的话	而已	根据	何况	么
还有	己	经	自身	倘	尔后	跟	何时	每
换句话说	既	经过	综上所述	倘或	反过来	因为	嘿	每当

3.3.2 实验结果

将分好词的中文语料作为的输入文件并指定合适的训练参数，即可进行中文词向量的训练。

TF-IDF 频率指数

表 **4**　测试词向量 **(**左**)** 和训练词向量 **(**右**)**

(文档标号, 词的标号)	词频	(文档标号, 词的标号)	词频
(0, 285)	1.0	(1, 285)	1.0
(1, 322)	1.0	(2, 255)	1.0
(3, 1680)	0.6642822	(3, 322)	1.0
(3, 215)	0.7474818	(7, 1680)	0.6702661
(4, 347)	1.0	(7, 215)	0.7421208
(6, 1680)	0.5684923	(8, 325)	0.7222694
(6, 266)	0.8226885	(8, 324)	0.6916118
(8, 270)	1.0	(9, 347)	1.0
(10, 996)	1.0	(12, 1277)	0.6699352
...	...	...	...
(4013, 255)	0.5998378	(8042, 258)	1.0
(4014, 1825)	0.7036199	(8043, 1832)	0.5452354
(4014, 266)	0.7105765	(8043, 1679)	0.6189684
(4015, 1763)	0.7195654	(8043, 325)	0.5653285
(4015, 289)	0.6944246	(8044, 1763)	0.7005996
(4016, 914)	0.7577426	(8044, 256)	0.7135545
(4016, 306)	0.6525535	(8045, 286)	1.0
(4017, 280)	1.0	(8047, 310)	1.0
(4018, 258)	1.0	(8048, 371)	1.0
(4019, 1763)	0.6969037	(8049, 492)	0.82337509
(4019, 256)	0.7171646	(8049, 289)	0.5674975
(4021, 371)	1.0	(8050, 269)	1.0
(4022, 269)	1.0	(8051, 534)	1.0

朴素贝叶斯分类预测：

```
test_corpus_seg/环境保护/U0008238.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000825.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008254.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008281.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008290.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008322.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008359.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000838.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008380.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008392.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008417.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008435.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008470.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008477.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U0008484.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000854.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000865.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000871.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000890.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U00090.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000903.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000935.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000940.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000956.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000959.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000966.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000974.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000995.txt : 实际类别：环境保护  -->预测类别：环境保护
test_corpus_seg/环境保护/U000999.txt : 实际类别：环境保护  -->预测类别：环境保护
预测完毕!!!
精度:0.9230
召回:0.9227
F1-score:0.9226
准确率:92.2714%

Process finished with exit code 0
```

图 **3**　预测结果

结果分析：我们建立关于留言内容的 TF-IDF 的朴素贝叶斯文本分类模型，通过改进的 TF-IDF 算法提取文本特征词，并根据属性之间的依赖关系添加隐藏的父节点，增强了属性之间的依赖关系，提高了情感分类的准确性。我们的模型正确率为 92.27%, 错误率仅为 7.73%。结合算法特点我们有理由相信在处理大规模数据集的时候，算法能表现出较好的特性，能较好的帮助工作人员需按照分类标签体系对政务平台的群众留言进行分类，根据相对应的部门职能，将该类信息分派至工人员手中。而且可以更改程序可以把错误的预测显示出来，将极大提高工作效率，以及降低工作差错。

3.4 评价指标

本模型采用的评价指标主要包括准确率以及 F1-Score。准确率 (Accuracy) 即对于给定的测试数据集，模型正确分类的样本数与总样本数之比。准确率在某些场合下确实可以有效地评价一个模型的好坏，然而在一些极端情况下，却显得不是那么重要了。例如：某个地区的总人数为 100 万人，而人群中患有某种病的人数只有 100 人。一个负责检测行人是否患有该病的模型只需要持续将行人归为“不患病”一类，即可获得超过 99% 的概率。

$$Accuracy = \frac{\text{“预测正确”的样本数}}{\text{总样本数}} \tag{16}$$

显然单纯使用准确率是不够的，我们引入 F1-Score 来解决上述现象，作为我们判断模型好坏的另一个指标。F1-Score 是精确率 (precision) 和召回率（recall）的调和函数。为了符合赛题评分标准，作出如下修改：

$$F1 - Score = \frac{\text{“预测标签为 1 且真实标签也为 1”的样本数}}{\text{“预测标签为 1”的样本数} + \text{“真实标签为 1”的样本数}} \tag{17}$$

评价结果：

通过 F1-Score 评价，我们得到了以下实验结果：

- F1-precision: 0.9238
- F1-recall: 0.9227
- F1-score:0.9226
- Accuracy:92.2714%

结果分析：F1 分数认为召回率和精确率同等重要，所以我们采用的是 F1-Score。当然希望检索结果 Precision 越高越好，同时 Recall 也越高越好，但事实上这两者在某些情况下有矛盾的。比如极端情况下，我们只搜索出了一个结果，且是准确的，那么 Precision 就是 100%，但是 Recall 就很低；而如果我们把所有结果都返回，那么比如 Recall 是 100%，但是 Precision 就会很低。因此在不同的场合中需要自己判断希望 Precision 比较高或是 Recall 比较高. 而我们的模型表现出较高的准确率。特别地，我们的模型获得了最高的 F1-Score 分值，也就是说我们的模型在不仅能够处理好匹配情况，还能在面对正面样本少于负面样本的情况下，准确地筛选出所需的正面样本。所以本文提出的模型得到了优秀的表现。

四、 文本相似度热度统计

针对群众反映的问题。我们使用 jieba+gensim 技术的将群众所反映的特定地点特定人群的热点问题按照一定标准进行评估和整理，及时整理出热点问题，有助于相关人员

高效做出应对方案。

4.1 模型的基础

4.1.1 Gensim

gensim 是 Radim Rehurek 写的一个用来处理文本相似度的 python 库。可以很方便的用 TF-IDF，LDA，LSI，word2vec 等模型，涵盖了 NLP 里常见的词袋模型，主题模型，词嵌入等

4.1.2 潜在语义索引 (LSI)

潜在语义索引 (Latent Semantic Indexing, 以下简称 LSI)，有的文章也叫 Latent Semantic Analysis(LSA)。其实是一个东西，后面我们统称 LSI，它是一种简单实用的主题模型。LSI 是基于奇异值分解 (SVD) 的方法来得到文本的主题的。

SVD: 对于一个 $m \times n$ 的矩阵 A, 可以分解为下面三个矩阵：

$$A_{m\times n} = U_{m\times n} \sum_{m\times n} V^T_{n\times n} \tag{18}$$

有时为了降低矩阵的维度到 k,SVD 的分解可以近似的写为：

$$A_{m\times n} \approx U_{m\times k} \sum_{k\times k} V^T_{k\times n} \tag{19}$$

如果把上式用到我们的主题模型, 则 SVD 可以这样解释：我们输入的有 m 个文本，每个文本有 n 个词。而 A_{ij} 则对应第 i 个文本的第 j 个词的特征值，这里最常用的是基于预处理后的标准化 TF-IDF 值。k 是我们假设的主题数，一般要比文本数少。SVD 分解后,U_{il} 对应第 i 个文本和第 l 个主题的相关度。V_{jm} 对应第 j 个词和第 m 个词义的相关度。$\sum_{lm}$ 对应第 l 个主题和第 m 个词义的相关度。

也可以反过来解释：我们输入的有 m 个词，对应 n 个文本。而 A_{ij} 则对应第 i 个词档的第 j 个文本的特征值，这里最常用的是基于预处理后的标准化 TF-IDF 值。k 是我们假设的主题数，一般要比文本数少。SVD 分解后，U_{il} 对应第 i 个词和第 l 个词义的相关度。V_{jm} 对应第 j 个文本和第 m 个主题的相关度。。$\sum_{lm}$ 对应第 l 个词义和第 m 个主题的相关度。

这样我们通过一次 SVD，就可以得到文档和主题的相关度，词和词义的相关度以及词义和主题的相关度。

LSI 简单实例

这里举一个简单的 LSI 实例，假设我们有下面这个有 11 个词三个文本的词频 TF 对应矩阵这里我们没有使用预处理，也没有使用 TF-IDF，在实际应用中最好使用预处理

后的 TF-IDF 值矩阵作为输入。我们假定对应的主题数为 2，则通过 SVD 降维后得到的三矩阵为：

$$\mathbf{U} \approx \mathbf{U_k} = \begin{bmatrix} -0.4201 & 0.0748 \\ -0.2995 & -0.2001 \\ -0.1206 & 0.2749 \\ -0.1576 & -0.3046 \\ -0.1206 & 0.2749 \\ -0.2626 & 0.3794 \\ -0.4201 & 0.0748 \\ -0.4201 & 0.0748 \\ -0.2626 & 0.3794 \\ -0.3151 & -0.6093 \\ -0.2995 & -0.2001 \end{bmatrix} \quad \mathbf{k} = 2 \quad \Sigma \approx \Sigma_{\mathbf{k}} = \begin{bmatrix} 4.0989 & 0.0000 \\ 0.0000 & 2.3616 \end{bmatrix}$$

$$\mathbf{V} \approx \mathbf{V_k} = \begin{bmatrix} -0.4945 & 0.6492 \\ -0.6458 & -0.7194 \\ -0.5817 & 0.2469 \end{bmatrix} \quad \mathbf{V}^\top \approx \mathbf{V}_\mathbf{k}^\top = \begin{bmatrix} -0.4945 & -0.6458 & -0.5817 \\ 0.6492 & -0.7194 & 0.2469 \end{bmatrix} \tag{20}$$

$$\begin{array}{c} \text{Terms} \\ \downarrow \\ \text{的} \\ \text{小区} \\ \text{被} \\ \text{年} \\ \text{施工} \\ \text{邮政} \\ \text{违规} \\ \text{请求} \\ \text{什么} \\ \text{街} \\ \text{路} \end{array} \quad \mathbf{A} = \begin{array}{c} \begin{array}{ccc} d1 & d2 & d3 \\ \downarrow & \downarrow & \downarrow \end{array} \\ \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 1 & 1 \end{bmatrix} \end{array} \quad \mathbf{q} = \begin{array}{c} q \\ \downarrow \\ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} \end{array}$$

从矩阵 U_k 我们可以看到词和词义之间的相关性。而从 V_k 可以看到 3 个文本和两个主题的相关性。大家可以看到里面有负数，所以这样得到的相关度比较难解释。

在上面我们通过 LSI 得到的文本主题矩阵可以用于文本相似度计算。而计算方法一般是通过余弦相似度。比如对于上面的三文档两主题的例子。我们可以计算第一个文本

和第二个文本的余弦相似度如下：

$$sim(d_1, d_2) = \frac{(-0.4945) * (-0.6458) + (0.6492) * (-0.7194)}{\sqrt{(-0.4945)^2 + 0.6492^2} + \sqrt{(-0.6458)^2 + (-0.7194)^2}} \tag{21}$$

4.2 Word2vec 基本原理

Word2Vec 是由 Google 的 Mikolov 等人提出的一个词向量计算模型。

- 输入：大量已分词的文本
- 输出：用一个稠密向量来表示每个词

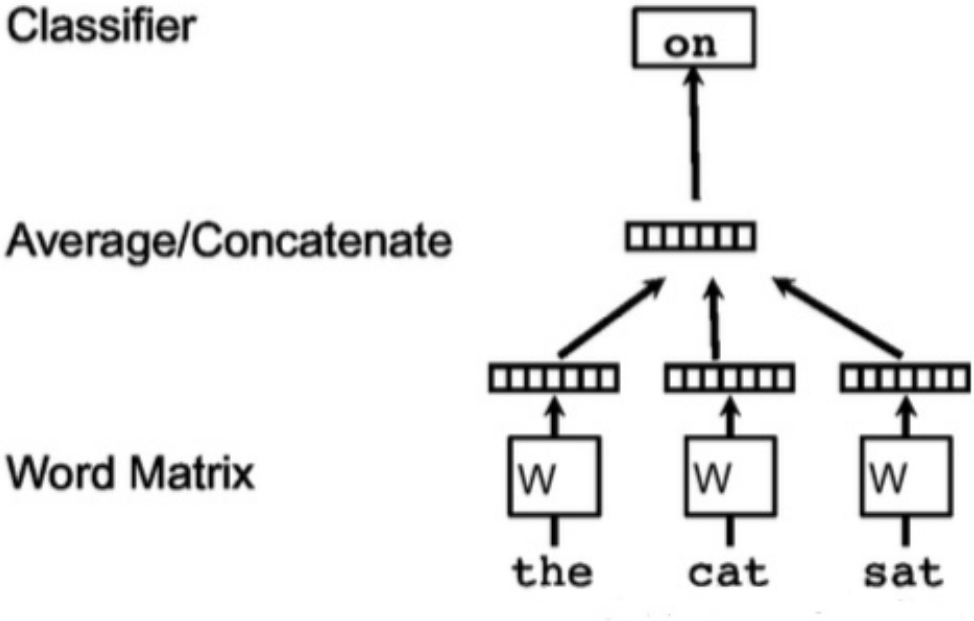

图 **4**　结构图 **(**来自文献 **[10])**

其中，每个单词都被映射到向量空间中，将上下文的词向量级联或者求和作为特征，预测句子中的下一个单词。一般地：给定如下训练单词序列, 目标函数是：

$$\frac{1}{T}\sum_{t=k}^{T-K} \log p(\omega_t|\omega_{t-k}, \cdots, \omega_t + k) \tag{22}$$

当然，预测的任务是一个多分类问题，分类器最后一层使用 softmax，计算公式如下：

$$p(\omega_t|\omega_{t-k}, \cdots, \omega_t + k) = \frac{e^y \omega_t}{\sum_i e^{y_i}} \tag{23}$$

这里的每一个可以理解为预测出每个的概率。因为在该任务中，每个词就可以看成一个类别。计算的公式如下：

$$y = b + Uh(\omega)_{t-k}, \cdots, \omega_{t+k}; W) \tag{24}$$

这里 U 和 b 都是参数，h 是将 $(\omega)_{t-k}, \cdots, \omega_{t+k})$ 级联或者求平均。因为每个单词都是一类，所以类别众多，在计算 softmax 归一化的时候，效率很低。因此使用 hierarical softmax 加快计算速度，其实就是 huffman 树。

4.3 Doc2vec 基本原理

训练句向量的方法和词向量的方法非常类似。训练词向量的核心思想就是说可以根据每个单词的上下文预测，也就是说上下文的单词对是有影响的。那么同理，可以用同样的方法训练 doc2vec。例如对于一个句子 i want to drink water，如果要去预测句子中的单词 want，那么不仅可以根据其他单词生成 feature，也可以根据其他单词和句子来生成 feature 进行预测。因此 doc2vec 的框架如下所示：

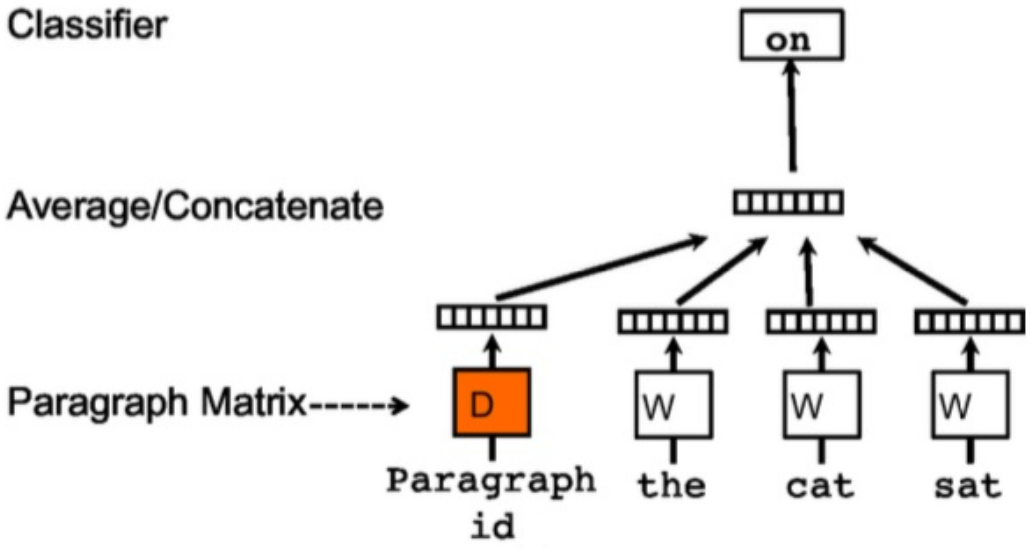

图 **5** **doc2vec** 框架图

每个段落/句子都被映射到向量空间中，可以用矩阵的一列来表示。每个单词同样被映射到向量空间，可以用矩阵的一列来表示。然后将段落向量和词向量级联或者求平均得到特征，预测句子中的下一个单词。

这个段落向量/句向量也可以认为是一个单词，它的作用相当于是上下文的记忆单元或者是这个段落的主题，所以我们一般叫这种训练方法为 Distributed Memory Model of Paragraph Vectors(PV-DM)

在训练的时候我们固定上下文的长度，用滑动窗口的方法产生训练集。段落向量/句向量在该上下文中共享。

总结 doc2vec 的过程, 主要有两步:

1. 训练模型，在已知的训练数据中得到词向量, softmax 的参数和, 以及段落向量/句向量
2. 推断过程（inference stage），对于新的段落，得到其向量表达。具体地，在矩阵中添加更多的列，在固定,, 的情况下，利用上述方法进行训练，使用梯度下降的方法得到新的 D, 从而得到新段落的向量表达

4.4 模型算法的实现

1. 输入文件是 excel，首先通过 pandas 获取 excel 信息，通过 jieba 分词进行处理，jieba 分词要首先自定义词典以及排除信息，这样效果会差异很大，然后形成一个二维数组。
2. 使用 gensim 中的 corpora 模块，将分词形成后的二维数组生成词典
3. 将二维数组通过 doc2bow 稀疏向量，形成语料库
4. 刚开始使用 TF 模型算法，后来更改为：LsiModel 模型算法，将语料库计算出 Tfidf 值。
5. 获取词典 token2id 的特征数
6. 计算稀疏矩阵相似度，建立一个索引
7. 读取 excel 行数据，通过 jieba 进行分词处理
8. 通过 doc2bow 计算测试数据的稀疏向量
9. 求得测试数据与样本数据的相似度

4.5 热度评价指标

对热点问题的挖掘分完类后，我们根据反应同种问题的数量和时间范围的长短，定义了热度评价指标 H，根据题目意思可以得出，热度评价指标 H 与反应同种问题的数量成正比，与时间范围的长短成反比，于是我们得到热度评价指标 H 的公式

$$H = \alpha_1 X_1 + \frac{\alpha_2}{X_2} \tag{25}$$

其中 X_1 为反应同种问题的数量，X_2 为时间范围的长短并以天为单位。我们假设 $alpha_1 = 0.5, \alpha_2 = 180$, 则可以对分类问题的热度评价指标进行计算并排序。

4.6 模型的实验结果

根据我们的热度评价指标分析结果，我们取前 5 个重点讨论的问题为热点问题：

1. A 市伊景园滨河苑捆绑销售车位
2. 丽发新城小区旁建搅拌厂严重扰民
3. A5 区劳动东路魅力之城小区油烟扰民

4. A3 区中海国际社区空地夜间施工噪音太大
5. A 市 A1 区同鑫家园被淹房屋

热度排名	问题ID	热度指数	时间范围	地点/人群	问题描述
1	1	31.71	2019/07/07至2019/09/01	A市伊景园滨河苑	A市伊景园滨河苑捆绑销售车位
2	2	31.23	2019/11/02至2020/01/06	A市丽发新城小区	丽发新城小区旁建搅拌厂严重扰民
3	3	16.73	2019/07/21至2019/09/25	A5区劳动东路魅力之城小区	A5区劳动东路魅力之城小区油烟扰民
4	4	7.05	2019/07/04至2019/11/22	A3区中海国际社区	A3区中海国际社区空地夜间施工噪音太大
5	5	3.72	2019/2/14至2019/7/11	A1区同鑫家园小区	A市A1区同鑫家园被淹房屋

图 **6**　热点问题表

问题ID	留言编号	留言用户	留言主题	留言时间	留言详情	点赞数	反对数
1	188801	A909180	投诉滨河苑针对广铁职工购房的霸王规定	2019-08-01 00:00:00	尊敬的张市长，您好！我叫李建义，来自湖北仙桃，虽然已经在广铁集团A市公	0	0
1	190337	A00090519	关于伊景园滨河苑捆绑销售车位的维权投诉	2019-08-23 12:22:00	投诉伊景园.滨河苑开发商捆绑销售车位！A市武广新城片区下的伊景园.滨河	0	0
1	191001	A909171	A市伊景园滨河苑协商要求购房同时必须购买车位	2019-08-16 09:21:33	商品房伊景园滨河苑项目是由A市政府办牵头为广铁集团铁路职工定向销售的楼	1	12
1	192739	A909188	请政府救救广铁集团的职工吧	2019-09-01 20:32:26	实在搞不懂买个单位福利房-伊景园滨河苑这么麻烦，认购都都认购了好几次，从	0	0
1	195511	A909237	车位捆绑违规销售	2019-08-16 14:20:26	对于伊景园滨河苑商品房，A市广铁集团违规捆绑车位销售至今，买房必须买车位	0	0
1	195995	A909199	关于广铁集团铁路职工定向商品房伊景园滨河苑项目的问题	2019-08-10 18:15:16	尊敬的市政府领导，您好！我是广铁集团基层职工，我要反应的问题是关于广铁集	0	0
1	196264	A00095080	投诉A市伊景园滨河苑捆绑车位销售	2019/8/7 19:52:14	A市伊景园·滨河苑现强制要求购房者捆	0	0
1	200085	A00010423	A市市政建设开发有限公司对广铁职工住宅项目的	2019-08-19 11:34:11	我是广铁的一名职工，对于A市市政建设开发有限公司操作广铁职工的组景同	2	9

图 **7**　部分热点问题留言明细表

五、 留言回复质量评价模型

5.1 基于 TextRank 的留言回复质量的关键词提取

本模型对于留言回复质量的关键词提取采用 python 的 jieba 工具包调用 TextBank 方法实现。即:jieba.analyse.textbank(sentebce, toK=200, withWeight=True,allowPOS=('nt','nz','n')),把，名类词作为限制词性，共现窗口默认为 5，我们选取了 TOP 词 200 制作了词云图，

如图表示文字越大表示该关键词的重要程度越高，比较突出的关键词为：问题、进行、工作、情况、您好、网友、回复、反映、如下、我们、收悉、相关、建设、感谢您、关于、支持、有关、项目、留言、根据。均是针对留言回复质量的方面的词语。

图 **8**　词频统计图

针对关键词的提取结果，可以归纳 (部分参考文献 [13]) 为下表所示，可以看出留言回复质量的一些要素。

表 **5**　关键词归纳结果

关键词	关键词归纳
依法依规、[政府发文][2014]247 号	可解释性
问题、进行、收悉、相关、建设、项目、有关、根据	相关性
我们	客观性
您好、网友、感谢您、支持、收悉	友好性

此外我们还应该考虑一些影响回复质量的因素，如信息贴近度，反馈信息的完备性（就提问每一个关键点的相关性大于某个阈值的频数概率），关键语句符合标准用语，回复的时效性（留言时间和答复时间不同）。

5.2 留言回复质量指标体系的确立

基于对留言回复质量要素的数据，根据提取出来的留言质量关键词对初始指标体系进行调整，最终确定了 5 维度 7 个指标的留言回复质量评价指标体系，如图所示。

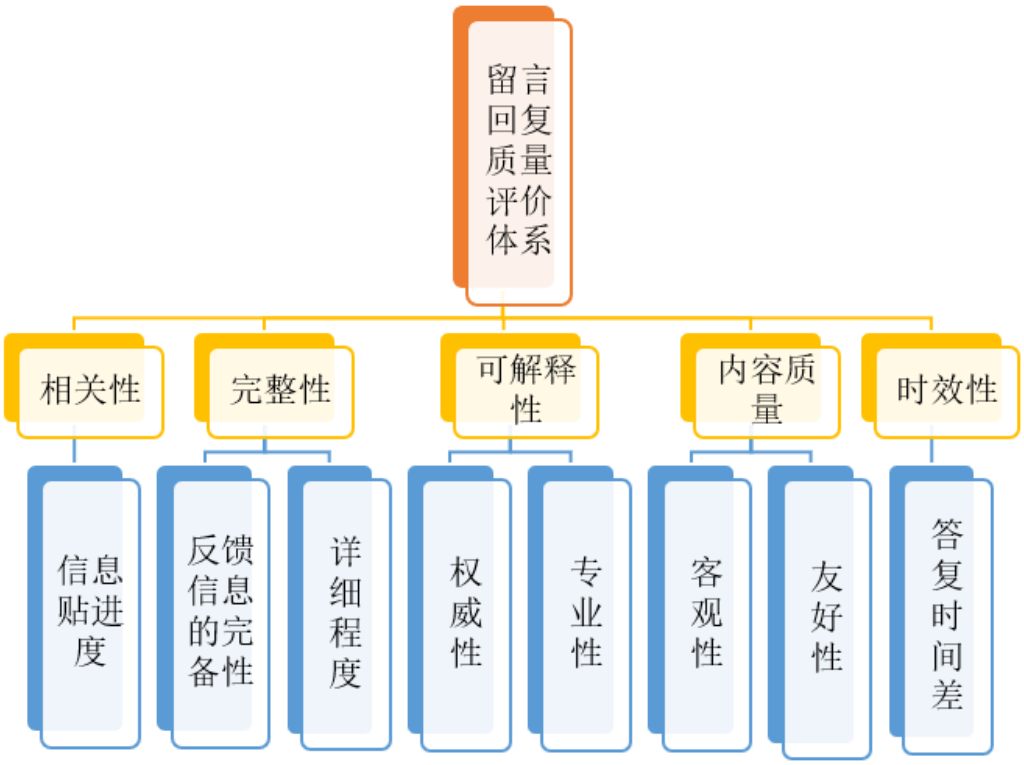

图 **9**　回复质量评价指标体系图

5.3 模型构成

对于网络留言的回复内容构建基于文本语义的评价模型，可以自动化地量化评估留言回复内容的质量情况。质量评价模型主要由 5 个一级指标和 8 和二级指标确定：留言回复的相关性、完整性和可解释性和及时性。由于原始数据没有给出 label 信息，因此本模型采取无监督方法，对于各个影响评价的子模块进行加权求和，最后得到关于每一条回复内容的质量得分。模型形式化如下：

$$QMR = \sum_{i=1}^{s} u_i \sum_{j=1}^{m_i} w_{ij} * Score_{ij} \tag{26}$$

在上式中：

$\boldsymbol{QMR}$：留言回复质量的得分；

$\boldsymbol{u_i}$：第 i 个一级指标的权重；

$\boldsymbol{s}$：一级指标的个数；

$\boldsymbol{w_{ij}}$：第 j 个二级指标相对于第 i 个一级指标的权重；

$\boldsymbol{m_i}$：第 i 个一级指标下的二级指标个数；

$\boldsymbol{Score_{ij}}$：第 i 个一级指标下的第 j 个二级指标的得分，即二级指标对留言回复质量的数值

5.4 基于 AHP 的指标权重计算

5.4.1 基于指标归类的两两判断矩阵

与以往采用人工主观判断各因素的相对重要程度，然后再去构造两两比较判断矩阵的方式不同，本文采用的是利用留言回复的指标归类结果，即每个二级指标所拥有的单元数量以及一级指标所拥有的单元数量，通过两两比较来构建判断矩阵。某个单元是从真实的数据中抽取出来的，是影响留言回复质量的重要因素。因此，本文采用的这种方法不仅仅避免了个人主观性因素，还以留言质量为导向进行定量的构造判断矩阵，使得决策结果更加可信。

若比较 n 个因子 $X = x_1, x_2, \cdots, x_n$ 对某因素 Z 的影响大小，可采用两两比较建立比较判别矩阵 $A = (a_{ij})_{n\times n}$, x_i 与 x 对 z 的影响之比为 a_{ij}, 反之，x 与 x_i 的影响之比为 $a_{ji} = 1/aij$. 当判断矩阵 A 满足如下性质时，称该矩阵具有完全一致性:$a_{ij} = 1, a_{ji} = 1/aij, a_{ij} > 0, a_{ij} = a_{ik}/a_{jk}$. 当矩阵达到完全一致时，其判断结果是相当准确的。而现实生活中，考虑到评价者自身知识、经验的有限以及处理事情的复杂度, 做到构建的判断矩阵完全一致难度很高。因此，在使用判断矩阵做决策前，需要使用一致性检验以保证结果的可靠性。

5.4.2 权重向量及一致性检验

对于 n 阶两两比较判断矩阵 $A = (a_{ij})_{n\times n}$, 使用和积法可近似的求矩阵特征向量，而本文中则利用 python 的 numpy 包的 linalg.eig() 函数直接计算出矩阵的特征值和特征向量，并取出最大的特征值 λ_{max} 和对应的归一化后的特征向量 $\omega = (\omega_1, \omega_1, \cdots, \omega_n)$ 即权重向量。

然后计算一致性指标 CI：

$$CI = \frac{\lambda_{max} - n}{n - 1} \tag{27}$$

当 $\lambda_{max} = n, CI = 0$, 意味着判断矩阵为完全一致矩阵，判断结果非常准确，而 CI 越大表示判断矩阵的完全一致性越差，通常认为 $CI < 0.1$ 时可接受判断矩阵的一致性，否则需

要对判断矩阵的元素赋值重新考虑。但是在判断矩阵的维度比较高时，原来的一致性检验方法的效果就不太准确了，需要有一种针对高纬度矩阵的一致性检验方法，本文的主要方法是引入修正值 RI。当修正后的指标 CR，见如下公式，当若 $CR = CI/RI < 0.1$，则认为此两两判断矩阵通过一致性检验。

$$CR = \frac{CI}{RI} \tag{28}$$

5.4.3 指标权重的计算

1. 构造判断矩阵 $A = (a_{ij})_{n\times n}$ 其中 a_ij 表示某个层次因素 i 和因素 j 相对于目标重要值;
2. 用求解矩阵特征值的方法解出最大特征值和特征向量，归一化后的特征向量即为权重;
3. 对判断矩阵的逻辑性进行一致性检验，若 $CR = CI/RI < 0.1$，则该矩阵通过一致性检验，认为该矩阵是可接受的。由此得到最终的评价指标权重如表所示

表 **6**　留言回复信息质量评价指标

一级指标	权重	二级指标	权重
相关性	0.1862	信息贴进度	1
完整性	0.1966	反馈信息的完备性	0.6543
		详细程度	0.3457
可解释性	0.4711	权威性	0.5190
		专业性	0.4810
内容质量	0.0662	客观性	0.6738
		友好性	0.3262
时效性	0.0799	答复时间差	1

5.5 子模型介绍

5.5.1 相关性评分模型

信息贴进度

留言回复质量的优劣,其中一个重要的因素是回复是否跟留言的信息贴进度比较高。模型采用基于语义的相关性算法来量化问答对之间的信息贴进度大小，该算法可以有效捕捉文本背后隐含的深层次语义距离，从而挖掘出更多有意义的文本信息。word2vec 可以把 word 用一个低维稠密向量表示，能够解决 one-hot 编码造成的维数灾难和语义相关问题。训练过程可以看作通过神经网络训练 N-gram 语言模型，并在训练过程中求出 word 对应的词向量。对留言语料库进行去停用词、分词预处理，构建并训练 word2vec 词向量模型，得到语料库中每个词汇的稠密词向量，用该词向量表示当前词语的语义信息。然后对留言详情和答复意见中各个词语进行向量拼接形成文档向量，两个文档向量计算余弦相似度可以得到文本信息贴进度函数。

$$F_1 = WORDSIM(\omega_i, \omega_{pj}) = \frac{\sum_i^n (x_{i1} \times x_{i2})}{\sqrt{\sum_i^n x_{i1}^2} \times \sqrt{\sum_i^n x_{i2}^2}} \tag{29}$$

5.5.2 完整性评分模型

反馈信息的完备性

留言回复的每一个关键点的相关性大于某个阈值的概率 F_2。

详细程度

基于经验知识，一般来说留言回复内容的详细程度与回复的文本长度有直接的关系。简短内容的回复信息量一般不够，评分应该较低；同时，较长文本的回复评分不应该过高。基于上述考量，本模型使用对数函数来量化回复的文本长度与评分的关系，建立完整性评分模型。模型形式化如下：

$$F_3 = \frac{1}{N_0} \sum_{i-1}^{N_0} \log_m L_i \tag{30}$$

其中，$F(x_i)$ 表示回复的完整性得分，L_i 表示句子的文本长度。

5.5.3 可解释性评分模型

权威性

权威性即留言回复中是否出现相关的标准文献，如若引用了相关的标准文件，则代表此条回复权威较高，对此加上相应的权重 F_4。

专业性

专业性，即关键词语符号专业用语，留言回复的内容需要有严密的逻辑和准确的表达，即回复信息是否有理有据，具有可解释性。假设当前留言回复文本中共有 N 个句子，将引用相关专业词典时出现的关键词和回复中每个句子的文本进行关键词匹配。若在回复的文本中匹配到相应的关键字，则认为该条回复引用了专业回答。统计回复信息

引用专业回答的句子数目 M，计算出现频率，即 M 与 N 的比值。使用该频率值量化可解释性评分。模型形式化如下：

$$F_5 = \frac{N}{M} \tag{31}$$

5.5.4 回复内容质量模型

友好性

政府留言回复的友好性是十分重要的，它代表着政府的形象，对于留言回复中，出现“您好，首先感谢您对我们工作的信任和支持”，“感谢您对我们工作的支持、理解与监督!”等语句代表答复友好。若在回复的文本中匹配到相应的语句，则认为该条回复友好。假设当前留言回复文本中共有 G 个句子，统计出现友好的句子数目 H，计算出现频率，即 G 与 H 的比值。使用该频率值量化友好性评分。模型形式化如下：

$$F_6 = \frac{G}{H} \tag{32}$$

客观性

留言回复问题是除友好语句是否出现，“我”，“我们”等第一人称词，如果出现则代表有个人主观色彩，对此扣除相应的权重 F_7。

5.5.5 及时性模型

留言回复的及时性也是衡量留言回复质量的一个重要的指标。即对于留言的答复，回复的越及时，评价越高。P 答复时间，Q 为留言时间，α 为加权系数, 我们规定 7 天以内回复得分为 100 分，7 天之后回复的分数 $F = \frac{7}{\text{天数}} \times 100$, 对此我们建立留言回复及时性函数:

$$F_8 = \alpha(P - Q) \tag{33}$$

5.6 模型的实现

1. 数据清洗，剔除“留言详情”或“答复意见”中文本长度小于 K 的数据。
2. 文本分词，对每条“留言详情”和“答复意见”文本进行分词处理
3. 去停用词
4. 构建专业的关键词词典。
5. 构建词向量模型，搭建 word2vec 模型并训练词向量
6. 搭建相关性评分模型、完整性评分模型、可解释性评分模型、内容质量频分模型、时效性评分模型。

7. 计算每条留言回复文本的质量得分。

5.6.1 建立评价等级

我们首先建立了答复意见质量论域 V,$V=(v_1,v_2,\cdots,v_8)$，答复质量 V=(很高，高，较高，中，较低，低，很低，差) 根据最后得到关于每一条答复意见的质量得分：

$$QMR=\sum_{i=1}^{s}u_i\sum_{j=1}^{m_i}w_{ij}*Score_{ij} \tag{34}$$

我们建立留言答复评价等级表格：

表 **7**　留言答复评价表

质量得分 $F(x)$	答复质量
$100\sim90$	很高
$90\sim80$	较高
$80\sim70$	高
$70\sim60$	中
$60\sim50$	低
$50\sim40$	较低
$40\sim30$	很低
$30\sim0$	差

5.6.2 实验结果

我们取了部分留言回复的内容，从相关性、完整性、可解释性、内容质量、时效性 5 个一级指标和信息贴进度等 8 个二级指标构建相关部门对留言的答复意见的评价模型。通过上述计算步骤，得到了部分留言回复质量对应评估等级结果如下：

留言编号		2549	2554	2555	2557	2574	2759
相关性	F1	0.756756	0.74352	0.62169	0.63502	0.7908	0.57974
完整性	F2	0.95939	0.85247	0.93926	0.74926	0.64926	0.67587
	F3	0.8411	0.8096	0.8242	0.8086	0.5627	0.5755
可解释性	F4	0	0	1	1	0	0
	F5	0.8256	0.7873	0.8538	0.5356	0.4149	0.616
内容质量	F6	0.33	0.67	0.4	0.33	0.67	0.67
	F7	1	1	1	1	1	1
及时性	F8	0.46667	0.5	0.5	0.5	0.46667	0.22581
QMR		0.80874	0.77955	0.85148	0.75843	0.64879	0.59991
答复质量		较高	高	较高	高	中	低

图 **10**　部分留言回复评估等级

从评价等级结果来看，留言回复的相关性，完整性，可解释性对答复的质量影响较大，留言回复的专业性、可靠性和权威性较好的留言 2549 和留言 2555 都得到了较高的答复质量。本文评价方法的优点是保留了原始信息的不确定性，采用证据组合的算法对具有模糊性以及不确定性的指标进行融合，能够很好地增强判断的客观性，降低主观性，提高真实性。给相关部门对留言的答复意见的评价提供了较好的评价方案。

六、 总结与后续研究建议

6.1 全文总结

1. 朴素贝叶斯模型发源于古典数学理论，有稳定的分类效率。本实验是对改进的 TFIDF 与隐朴素贝叶斯结合，对文本的情感进行分类。通过改进的 TF-IDF 算法提取文本特征词，并根据属性之间的依赖关系添加隐藏的父节点，增强了属性之间的依赖关系，提高了情感分类的准确性。
2. LSI 低维空间表示可以刻画同义词，同义词会对应着相同或相似的主题。降维可去除部分噪声。充分利用冗余数据。无监督/完全自动化。与语言无关。
3. LsiModel 模型算法它的算法原理很简单，一次奇异值分解就可以得到主题模型，同时解决词义的问题。
4. 留言回复质量模型从 5 个一级指标 8 个二级指标对客服的回复质量进行量化描述，通过利用留言回复的指标归类结果，即每个二级指标所拥有的单元数量以及一级指标所拥有的单元数量，通过两两比较来构建判断矩阵，得出权重。实验结果表明，该模型可以很好地评价的留言的答复意见。该模型研究也有助于建设智能问答的网络平台，促进优质和高效的问题回复，帮助公民咨询服务网站提供更好的用户体验。

。

6.2 后续研究建议

- **针对问题一：**朴素贝叶斯在大型的数据集上表现出来难得的速度和准确度, 而在训练集不是很大的情况下, 效果明显不如其它算法。通过实验分析, 发现出现这种现象的原因是: 在训练样本少的情况下, 某个生僻特征是否出现的随机性很大。而朴素贝叶斯公式是一种连乘积的值, 这使得这种特征在文本分类中起到了一种统治性的作用。如果扩大数据集，模型能进一步提升准确率，达到 95%，将错误率有效控制在 5% 以内。
- **针对问题二：**一般来说自定义词库、停用词和构建专业词典，取得得效果会更好更准确。LSI 的概率模型假设文档和词的分布是服从联合正态分布的，但从观测数据来看是服从泊松分布的。因此 LSA 算法的一个改进 PLSA 使用了多项分布，其效果要好于 LSA。
- **针对问题三：**社交媒体信息传播的速度、广度和深度对信息质量提出了更高的要求，政府部门的留言回复更应注重其信息质量是否为用户所认同和满意。以用户为中心、基于用户满意度的政府信息服务是目前主流的研究内容，政府社交媒体信息质量更应转向用户的价值所得，转向用户的具体感受，唯有如此，才能在互联网载体上推进国家治理体系与治理能力的现代化，体现用户需求变化和政府行为策略的高度交互。本文建立的评价体系，在方法上厘清政府留言回复的质量，有助于政府社交媒体信息质量的提升及为后续采纳策略提供政策参考。本研究只是一个基于政府留言回复考察的信息质量的评价检视，在实践运作中，如何有效分析各种类型留言的信息质量影响因素以及准确理解用户信息需求偏好，以提高政府社交媒体应用中的信息质量，是一个值得探究的长期性复杂工作，这将是后续的研究课题。

参考文献

[1] 安艳辉, 董五洲, 游自英. 基于改进的朴素贝叶斯文本分类研究 [J]. 河北省科学院学报,2007(01):22-25.

[2] 宋晓敏. 基于改进贝叶斯算法的中文信息分类研究 [D]. 北京邮电大学,2019.

[3] 李云帆, 胡皓程, 康佳乐. 朴素贝叶斯算法的应用 [J]. 电脑编程技巧与维护,2018(10):4-7+41

[4] 许甜华, 吴明礼. 一种基于 TF-IDF 的朴素贝叶斯算法改进 [J]. 计算机技术与发展,2020,30(02):75-79.

[5] 李晓东, 肖基毅, 邹银凤. 基于改进的 TF-IDF 与隐朴素贝叶斯的情感分类研究 [J]. 南华大学学报 (自然科学版),2019,33(02):79-84.

[6] 周练. Word2vec 的工作原理及应用探究 [J]. 科技情报开发与经济,2015,02:145-148.

[7] 郑文超, 徐鹏. 利用 word2vec 对中文词进行聚类的研究 [J]. 软件,2013,12:160-162.

[8] 顾益军, 樊孝忠, 王建华, 汪涛, 黄维金. 中文停用词表的自动选取 [J]. 北京理工大学学报,2005,04:337-340.

[9] 曹卫峰. 中文分词关键技术研究 [D]. 南京理工大学,2009.

[10] QuocLe.TomasMikolov,Distributed Representations of Sentences and Documents[J].Google Inc, 1600 Amphitheatre Parkway, Mountain View, CA 94043

[11] https://blog.csdn.net/john_xyz/article/details/79208564

[12] https://blog.csdn.net/weixin_40834089/article/details/82656472

[13] 杨开平, 李明奇, 覃思义. 基于网络回复的律师评价方法 [J]. 计算机科学,2018,45(09):237-242.