基于自然语言处理技术建立智慧政务系统

摘要

随着机器学习和自然语言处理技术的深入，让机器代替人工处理繁琐事宜渐渐成为主流。本文的目的是基于自然语言处理技术建立智慧政务系统，解决 3 个主要问题。问题 1 是群众留言分类，根据给出的数据，建立关于留言内容的一级标签分类模型。采取的方法是词袋模型与朴素贝叶斯分类法相结合。问题 2 是热点问题的挖掘，将某一时段内反应特定地点或特定人群问题的留言进行归类，定义合理的热度评价指标，给出评价结果，按格式给出排名前 5 的热点问题及留言信息。采取的方法是构建 LDA 模型以及热度指数。问题 3 是答复意见的评价。针对相关部门对留言的答复意见，从相关性、可解释性以及完整性的角度对答复意见的质量给出评价方案。

关键词：特征选择；词袋模型；朴素贝叶斯分类；LDA 模型；热度指标；同义词词林

Abstract

With the development of machine learning and natural language processing technology, it is becoming more and more popular to let machines do the tedious work instead of humans. The purpose of this paper is to build an intelligent government system based on natural language processing technology and solve three main problems. Question 1 is about the classification of comments from the masses. According to the data given, the first-level label classification model of comments is established. The method adopted is the combination of bag-of-words model and naive bayesian classification. Question 2 is the mining of hot issues. It classifies the comments reflecting the problems of specific places or people in a certain period of time, defines a reasonable heat evaluation index, gives the evaluation results, and presents the top 5 hot issues and message information in a format. The method adopted is to build LDA model and heat index. Question 3 is the evaluation of the replies. According to the comments of the relevant departments, the quality of the comments is evaluated from the perspective of relevance, interpretability and completeness.

Key words: feature selection; bag-of-words model; Naive bayesian classification; The LDA model; Heat index; Forest of synonyms

目录

一、问题 1：群众留言分类

1.1 文本预处理

文本预处理包含中文分词和去停用词两个过程。

中文分词是将一个汉字序列切分成一个一个单独的词。现有的分词方法可分为三大类：词典分词方法、理解分词方法和基于统计的分词方法。词典分词方法，又称机械分词法，原理是根据一定的规则把待分析的文本与一个容量较大的机器词典存有的词条进行匹配，如果在词典中寻找出一个字符串，表明匹配成功就可以识别出这个词。理解分词方法，是让机器模拟人对语句的理解，进而达到识别字或词组的效果，其基本思想是在分词的过程中同时进行句法、语义分析，来处理歧义现象[1]。基于统计的分词方法是在给定大量已经分词的文本的前提下，利用统计机器学习模型学习词语切分的规律，从而实现对未知文本的切分。随着大规模语料库的建立，统计机器学习方法的研究和发展，基于统计的中文分词方法渐渐成为了主流方法。

中文分词就是让计算机系统按照一定规则在文本中添加空格等将其划分为字或词组。Jieba 提供了三种分词模式：

精确模式：试图将句子最精确地切开，适合文本分析；

全模式：把句子中所有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；

搜索引擎模式：在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。

在信息检索中，为节省存储空间和提高搜索效率，在处理自然语言数据（或文本）之前或之后会自动过滤掉某些字或词，这些字或词被称为停用词。例如，常见的“的”、“在”、“一个”等字词，均属于停用词。常用的停用词词库有哈工大停用词词库、百度停用词表等。根据不同的数据要求选择不同的停用词词库，可适当的添加、删除。

1.2 特征选择

文本预处理后以特征项集合的形式存在，此时特征项集合中的特征项数量非常多。如果直接使用，则会造成模型维数灾难且影响最终的分类效果。此外，并不是每个特征项对分类都有贡献，只有在特定的语义环境中，特征词才能体现应用价值。特征选择的过程就是文本语义规则化的过程，只有规则化的语义才能被计算机表示。因此，对特征项集合进行降维处理，选择对文本信息贡献度大的特征词是非常有必要的[2]。特征选择在数据挖掘、模式识别和机器学习等多个领域都得到了广泛的研究。

Jieba 提供了两种关键词提取方法，分别基于 TF-IDF 算法和 TextRank 算法。TF-IDF（词频-逆文件频率）是一种统计方法，用以评估一个词语对于一个文件集或一个语料库中的一份文件的重要程度，其原理可概括为：一个词语在一篇文章中出现的次数越多，同时在所有文档中出现次数较少，越能代表该文章。TF-IDF = TF*IDF。

TF：词频，某一个给定的词语在该文件中出现的次数，计算公式为 $TF_w = \frac{在某一类中词条w出现的次数}{该类中所有的词条数目}$。

IDF：逆文件频率，如果包含词条的文件越少，则说明词条具有很好的类别区分能力，

计算公式为$IDF = \log\left(\frac{\textit{语料库的文档总数}}{\textit{包含词条}w\textit{的文档数}+1}\right)$。

1.3 词袋模型

文本表示是自然语言处理中的基础工作，文本表示的好坏直接影响到整个自然语言处理系统的性能。文本表达中比较经典的理论是向量空间模型，在文本分析中也被称为词袋模型（Bag of Words）。BoW 模型最初被用在文本分类中，其基本思想是假定对于一个文本，忽略其词序和语法、句法，仅仅将其看作是一些词汇的集合，而文本中的每个词汇都是独立的。简单来说，就是将每篇文档都看成是一个袋子，然后根据袋子里装的词汇对其进行分类。举个例子：

文档 1：“我喜欢跳舞，小明也喜欢。”

文档 2：“我也喜欢唱歌。”

基于以上两个文档，构造一个由文档中的关键词组成的词典：dictionary = {1:“我”,2:“喜欢”,3:“跳舞”,4:“小明”,5:“也”,6:“唱歌”}。这个一共包含 6 个不同的词与，利用词典的索引号，上面两个文档每一个都可以用一个 6 维向量表示。根据各个文档中关键词出现的次数，便可以将上述两个文档分别表示成向量的形式：

文档 1：[1 2 1 1 1 0]

文档 2：[1 1 0 0 1 1]

根据词袋模型得出的文本向量可用于相似度计算和文本分类。

1.4 朴素贝叶斯分类法

贝叶斯分类法是统计学分类方法，它可以预测类隶属关系的概率。贝叶斯分类基于贝叶斯定理，即$P(B|A) = \frac{P(A|B)*P(B)}{P(A)}$,P(类别|特征)=P(特征|类别)*P(类别)/P(特征)。朴素贝叶斯分类法假定一个属性值在给定类上的概率独立于其他属性的值，这一假定被称为类条件独立性。朴素贝叶斯分类法的基本思想是如果一个事物在一些属性条件发生的情况下，事物属于 C_1的概率>属于 C_2的概率，则判定事物属于 C_1。

已知特征 F_1、F_2、F_3相互独立，以及在类别已知条件下出现特征 F_1、特征 F_2、特征 F_3的概率，则可知在特征 F_1、F_2、F_3条件下，归属于 C_1、C_2、C_3等类别的概率。

$$P(C_1|F_1F_2F_3) = \frac{P(F_1F_2F_3C_1)}{P(F_1F_2F_3)} = \frac{P(F_1F_2F_3|C_1)P(C_1)}{P(F_1F_2F_3)} = \frac{P(F_1|C_1)P(F_2|C_1)P(F_3|C_1)P(C_1)}{P(F_1)P(F_2)P(F_3)}$$

$$P(C_2|F_1F_2F_3) = \frac{P(F_1|C_2)P(F_2|C_2)P(F_3|C_2)P(C_2)}{P(F_1)P(F_2)P(F_3)}$$

$$P(C_3|F_1F_2F_3) = \frac{P(F_1|C_3)P(F_2|C_3)P(F_3|C_3)P(C_3)}{P(F_1)P(F_2)P(F_3)}$$

要比较 $P(C_1|F_1F_2F_3)$、$P(C_2|F_1F_2F_3)$、$P(C_3|F_1F_2F_3)$等条件概率的大小，只需比较相应的分子大小。若 $P(C_1|F_1F_2F_3)$值最大，则根据特征 F_1、F_2、F_3，可将其归为 C_1类。

1.5 实验

1.5.1 数据准备

在对文本进行分词和去停用词前，先进行正则表达式。正则表达式是对字符串操作的一种逻辑公式，就是用事先定义好的一些特定字符、及这些特定字符的组合，组成一个“规则字符串”，这个“规则字符串”用来表达对字符串的一种过滤逻辑。对附件 2 中的留言详情进行处理，用空格代替详情中的数字、字母等，消除文本中出现的网址等不重要的字符串，从而更好的进行分词和去停用词。对于由于爬虫所导致的不间断空白符\xa0 和全角的空白符\u3000 使用字符串函数进行消除。

在该题中，对经过正则表达式处理后的文本进行 jieba 分词精确模式。所选用的停用词词典是“哈工大停用词词库”、“四川大学机器学习智能实验室停用词库”、“百度停用词表”等等停用词表的结合，比较完整全面。

1.5.2 特征选择

通过 jieba.analyse.extract_tags 方法可以基于 TF-IDF 算法进行关键词提取，topK 为返回几个 TF-IDF 权重最大的关键词，默认值为 20，在本题中每个文本提取 5 个关键词。示例数据的特征选取结果如图 1 所示：

```
路段 未管 文明城市 车流 安全隐患
护栏 立水 一方 电等 过往行人 锈渣斑斑
检测站 汽车 这家 环保局 无证
工地 民工 维权 态度恶劣 胡书记
丁字街 商户 形为 交通 城管局
摆出来 城管 慢慢 门口 上有政策
酒店 营业 损失 现县 蓝波
景观灯 人到 九亿 不好 黑黑的
公厕 石期 早厕 市要 蚊蝇
轨道交通 规划 磁浮 换乘 快线
轨道交通 规划 磁浮 换乘 快线
爱心卡 残疾人 地铁 乘坐 地跌
停电 每次 弓线 丽路 锦楚
高压线塔 用电 周边 保障 居民
停电 小区 线路 朝晖路 锦楚
供暖 近年 西地省 楚江 规划
供暖 室内 冬天 胡书记 潮湖
幸福感 市委 市政府 采暖 安全隐患
出行 请问 县要 公交线路 试运行
公交车 晚上 工业园 延迟 出行
监控 乘客 插卡 破旧不堪 车人
迎丰 公园 路灯 锻炼 熄灯
市民 公园 中坡山 遥感 禁罚
球场 一拖二 推三 连片 修好
雅礼洋湖 中学 家长 清理 教职工
英拌 后及 城边 脏乱 不堪
丽城 烟雾弥漫 垃圾场 担心 草木灰
小区 冯家冲 居民 消防 水源
珠山 从区 东楚桥 水有 镇政府
工程量 清单 标价 招标 空白
业主 物业 车位 收费 小区
小区 业主 物业公司 耀凯 栋楼
```

图 1 示例数据的特征选取结果

全部数据的特征选取结果如图 2 所示：

```
路段 未管 文明城市 车流 安全隐患
护栏 在水一方 电等 过往行人 锈迹斑斑
业主 物业 车位 收费 小区
不洗 华庭 水是 健康 霉味
不洗 华庭 水是 健康 霉味
小区 业主 物业公司 耀凯 栋楼
供暖 近年 西地省 楚江 规划
停水 自来水厂 胡书记 居民 小区业主
垃圾处理 小区 城市 物业公司 业主
小区 绿化带 业主 人管 社区
小区 绿化带 业主 人管 社区
居委会 业委会 业主 非法 滨江
自来水 溪湖 停水 用水 恳请
金晖 业主 交房 物业 限电
停电 每次 号线 丽路 锦楚
高压线塔 用电 周边 保障 居民
停电 小区 线路 朝晖路 锦楚
融城 城区 片区 金阳 发展
城市绿化 却显 滨水 城市道路 行道树
停电 跳闸 区楚府 雅苑 楚府
建造师 建望 证书 省辉 违法行为
黄谷路 嘉园栋 成户 居民 单元房
检测站 汽车 这家 环保局 无证
长株 磁悬浮 城铁 中速 对接
垃圾 围子 嘉华 露天 安置
磁悬浮 中速 西站 时速 规划
高铁 城铁 南站 黄花 机场
垃圾 围子 人行道 嘉华 市貌
供暖 室内 冬天 胡书记 潮潮
投诉无门 区银盆 华韵 租房 职能部门
幸福感 市委 市政府 采暖 安全隐患
沙岭 先锋 城铁 从观 原本
长株 板塘 城际 列车 市板
```

图 2 全部数据的特征选取结果

1.5.3 标签赋值

本题基于监督学习，给定已有的标签，附件 2 所给出的一级分类为“城乡建设”、“环境保护”、“交通运输”、“教育文体”、“劳动和社会保障”、“商贸旅游”以及“卫生计生”。对此分别用数字 1~7 表示，以便后续的分类。示例数据的赋值结果如图 3 所示：

```
all_class= [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4,
4, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,
5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5,
5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,
6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,
7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7]
```

图 3 数据赋值结果

1.5.4 词向量及分类

对经过 jieba 分词提取的关键字进行 BoW 模型，得到词向量，再对词向量进行朴素贝叶斯分类。示例数据的分类结果如图 4 所示：

```
result= [2, 2, 2, 2, 2, 2, 2, 2, 6, 1, 1, 2, 2, 2, 1, 6, 2, 2, 6, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2, 1, 6, 2, 6, 2, 2, 2, 2, 2, 1, 2, 2,
2, 2, 2, 2, 1, 2, 2, 2, 2, 7, 2, 6, 2, 2, 2, 2, 2, 6, 2, 6, 2, 2, 2, 2, 2, 2, 6, 2, 2, 6, 2, 2, 6, 2, 6, 2, 6, 2, 2, 2, 6, 2, 2, 6, 2, 2, 2,
2, 2, 2, 2, 6, 6, 2, 5, 1, 5, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 2, 4, 4, 4, 4, 4, 4, 4, 2, 4, 4, 4, 7, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 7, 4, 4, 4, 4, 4, 2,
4, 4, 4, 2, 4, 4, 4, 4, 4, 4, 4, 6, 7, 6, 2, 4, 2, 2, 4, 2, 4, 2, 6, 4, 7, 4, 4, 4, 2, 2, 2, 6, 2, 6, 6, 7, 2, 4, 4, 2, 4, 4, 4, 4, 2, 4, 4,
2, 2, 4, 2, 2, 7, 5, 5, 5, 5, 5, 7, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5,
5, 5, 5, 2, 5, 2, 5, 5, 6, 5, 5, 2, 5, 5, 5, 2, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5, 5, 2, 5, 5,
7, 5, 5, 5, 5, 5, 5, 5, 2, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,
6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,
7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7]
```

图 4 示例数据分类结果

全部数据的分类结果如图 5 所示：

result= [3, 3, 6, 7, 7, 1, 1, 1, 1, 1, 1, 1, 1, 6, 1, 1, 1, 3, 3, 1, 7, 2, 2, 1, 1, 1, 3, 1, 1, 3, 3, 1, 3, 3, 1, 1, 1, 1, 1, 3, 1, 1, 3,
1, 1, 5, 1, 4, 4, 1, 3, 1, 1, 1, 1, 1, 1, 6, 7, 1, 2, 1, 1, 1, 2, 1, 2, 1, 1, 1, 1, 1, 2, 1, 3, 7, 1, 3, 2, 1, 1, 1, 2, 3, 6, 1, 1, 1, 1,
1, 1, 1, 3, 1, 6, 1, 1, 3, 3, 7, 1, 3, 1, 1, 3, 3, 1, 1, 3, 1, 7, 3, 1, 2, 1, 3, 3, 3, 7, 3, 3, 1, 3, 6, 6, 1, 1, 3, 3, 1, 1, 6, 1, 2, 1,
6, 1, 1, 1, 1, 3, 1, 1, 1, 1, 7, 1, 6, 1, 6, 1, 1, 6, 1, 1, 1, 1, 1, 1, 1, 6, 1, 1, 3, 1, 1, 6, 1, 2, 4, 6, 1, 3, 1, 1, 1, 1, 1, 3, 1, 1,
1, 1, 1, 1, 3, 1, 1, 1, 4, 4, 3, 3, 3, 3, 3, 1, 1, 1, 3, 3, 3, 1, 1, 1, 1, 3, 2, 7, 3, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 5, 2, 3, 1, 1,
1, 1, 3, 1, 1, 2, 1, 1, 3, 3, 3, 3, 1, 3, 3, 1, 1, 3, 7, 3, 3, 3, 1, 3, 3, 2, 7, 7, 1, 1, 6, 1, 3, 3, 6, 1, 3, 6, 3, 3, 3, 3, 3, 3, 1, 3,
3, 3, 7, 3, 1, 1, 1, 3, 3, 3, 3, 3, 3, 3, 1, 1, 1, 1, 1, 1, 1, 1, 3, 7, 1, 1, 1, 1, 2, 1, 7, 5, 3, 1, 3, 3, 1, 3, 3, 3, 1, 2, 7, 3, 1, 3,
3, 1, 1, 1, 1, 1, 6, 3, 1, 3, 1, 3, 2, 3, 3, 1, 3, 1, 1, 3, 1, 2, 1, 1, 1, 3, 4, 3, 2, 7, 6, 1, 1, 1, 1, 1, 1, 2, 1, 1, 3, 2, 1, 3, 1, 6,
4, 1, 1, 7, 7, 1, 3, 5, 3, 3, 1, 7, 1, 5, 5, 1, 7, 5, 1, 1, 7, 1, 7, 5, 1, 1, 1, 3, 3, 1, 1, 1, 7, 3, 3, 7, 3, 3, 1, 1, 1, 1, 2, 3, 1, 6,
3, 1, 1, 1, 1, 3, 1, 6, 6, 3, 7, 3, 1, 1, 1, 5, 3, 1, 1, 1, 1, 1, 1, 1, 1, 1, 5, 5, 1, 2, 3, 3, 6, 1, 1, 1, 3, 1, 3, 3, 5, 1, 2, 2, 7, 3,
2, 1, 1, 1, 2, 7, 3, 5, 3, 1, 3, 7, 1, 1, 7, 3, 1, 1, 1, 5, 3, 1, 1, 1, 1, 1, 5, 3, 1, 1, 1, 2, 1, 1, 1, 3, 2, 3, 1, 2, 1, 1, 2, 3, 2, 1,
2, 1, 1, 2, 2, 2, 1, 3, 6, 1, 1, 1, 6, 1, 1, 7, 3, 1, 1, 1, 1, 2, 1, 2, 1, 3, 5, 1, 1, 3, 3, 1, 1, 7, 1, 1, 1, 7, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 3, 6, 1, 3, 1, 1, 1, 2, 3, 3, 3, 7, 3, 1, 3, 1, 1, 1, 1, 1, 4, 1, 1, 1, 1, 1, 1, 3, 3, 3, 3, 6, 1, 1, 1, 1, 1, 3, 1, 1,

图 5 全部数据分类结果

1.5.5 实验步骤

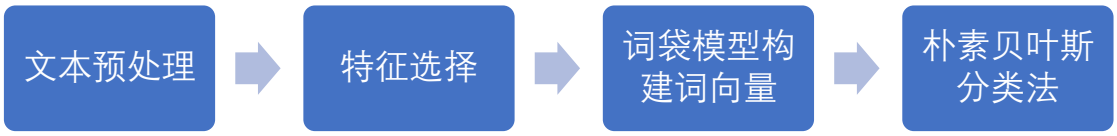

图 6 问题一实验步骤

1.5.6 评价指标

采用文本分类经常使用的评价指标作为实验结果的依据，即精确率、召回率以及 F-score 值。其中，精确率又称查准率，即正确预测为正的占全部预测为正的比例，用 P 表示；召回率又称查全率，即正确预测为正的占全部实际为正的比例，用 R 表示；F-score 是一种统计量，是在信息检索领域中较常见的评价标准，用于评价某个分类模型的好坏，用 F_1 表示[3]。

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

其中 P_i 为第 i 类的查准率，R_i 为第 i 类的查全率。

根据附件 2 示例数据得到的 F-score = 0.7093894879862296，根据全部数据得到的 F-score = 0.8459308864925978。当数据越多时，分类的效果越好。

二、问题 2：热点问题挖掘

2.1 预备知识

隐含狄利克雷分布（Latent Dirichlet Allocation，简称 LDA）是由 David M. Blei、Andrew Y. Ng、Michael I. Jordan 在 2003 年提出的，是一种词袋模型，它认为文档是一组词构成的集合，词与词之间是无序的。一篇文档可以包含多个主题，文档中的每个词都是由某个主题生成的，LDA 给出文档属于每个主题的概率分布，同时给出每个主题上词的概率分布。LDA 是一种无监督学习，在文本主题识别、文本分类、文本相似度计算和文章相似推荐等方面都有应用[4]。

2.1.1 gamma 函数

Gamma 函数的定义：

$$\Gamma(\mathrm{x}) = \int_0^{\infty} t^{x-1} e^{-t}\, dt$$

分部积分后，可以发现 Gamma 函数如有这样的性质:

$$\Gamma(\mathrm{x}+1) = x\Gamma(\mathrm{x})$$

Gamma 函数可以看成是阶乘在实数集上的延拓，具有如下性质:

$$\Gamma(\mathrm{n}) = (n-1)!$$

2.1.2 二项分布

二项分布是 N 重伯努利分布，即为 X ~ B(n，p). 概率密度公式为:

$$\mathrm{P}(\mathrm{K}=\mathrm{k}) = \binom{\mathrm{n}}{\mathrm{k}} p^k (1-p)^{n-k}$$

2.1.3 多项分布

多项分布, 是二项分布扩展到多维的情况. 多项分布是指单次试验中的随机变量的取值不再是 0-1 的，而是有多种离散值可能（1,2,3...,k）.概率密度函数为:

$$\mathrm{P}(x_1, x_2, \ldots x_k; n, p_1, p_2, \ldots, p_k) = \frac{n!}{x_1! \ldots x_k!} p_1^{x_1} \ldots p_k^{x_k}$$

2.1.4 Beta 分布

Beta 分布的定义：对于参数 alpha > 0，beta > 0，取值范围为[0，1]的随机变量 x 的概

率密度函数为：

$$\mathrm{f}(\mathrm{x};\alpha,\beta)=\frac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1}$$

其中，

$$B(\alpha,\beta)=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$$

2.1.5 共轭先验分布

在贝叶斯概率理论中，如果后验概率$\mathrm{p}(\theta|\mathrm{x})$和先验概率$\mathrm{p}(\theta)$满足同样的分布律，那么，先验分布和后验分布被叫做共轭分布，同时，先验分布叫做似然函数的共轭先验分布。

$$p(\theta|x)=\frac{p(\theta,x)}{p(x)}$$

Beta 分布是二项式分布的共轭先验分布，而狄利克雷(Dirichlet)分布是多项式分布的共轭分布。共轭的意思是，以 Beta 分布和二项式分布为例，数据符合二项分布的时候，参数的先验分布和后验分布都能保持 Beta 分布的形式，这种形式不变的好处是，我们能够在先验分布中赋予参数很明确的物理意义。

2.1.6 Dirichlet 分布

Dirichlet 的概率密度函数为：

$$f(x_1,x_2,\dots x_k;\alpha_1,\alpha_2,\dots,\alpha_k)=\frac{1}{B(\alpha)}\prod_{i=1}^{k}{x_i}^{\alpha^i-1}$$

其中，

$$B(\alpha)=\frac{\prod_{i=1}^{k}\Gamma(\alpha^i)}{\Gamma(\sum_{i=1}^{k}\alpha^i)},\sum_{i=1}^{k}x^i=1$$

根据 Beta 分布、二项分布、Dirichlet 分布、多项式分布的公式，可得 Beta 分布是二项式分布的共轭先验分布，而狄利克雷(Dirichlet)分布是多项式分布的共轭分布。

2.1.7 Beta / Dirichlet 分布的一个性质

如果 $p\ Beta(t|\alpha,\beta)$，则

$$E(p)=\int_0^1 t*Beta(t|\alpha,\beta)\,dt$$

$$=\int_0^1 t*\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}t^{\alpha-1}\,(1-t)^{\beta-1}dt$$

$$=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\int_0^1 t^{\alpha}\,(1-t)^{\beta-1}dt$$

上式右边的积分对应到概率分布$Beta(t|\alpha+1,\beta)$，对于这个分布，有

$$\int_0^1 \frac{\Gamma(\alpha+\beta+1)}{\Gamma(\alpha)\Gamma(\beta)} t^{\alpha}(1-t)^{\beta-1}dt = 1$$

把上式带入 E(p)的计算式，得到

$$E(p) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \cdot \frac{\Gamma(\alpha+1)\Gamma(\beta)}{\Gamma(\alpha+\beta+1)}$$
$$= \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha+\beta+1)} \cdot \frac{\Gamma(\alpha+1)}{\Gamma(\alpha)}$$
$$= \frac{\alpha}{\alpha+\beta}$$

这说明，对于对于 Beta 分布的随机变量，其均值可以用$\frac{\alpha}{\alpha+\beta}$来估计。Dirichlet 分布也有类似的结论，如果$\vec{p} \sim Dir(\vec{t}|\vec{\alpha})$，同样可以证明：

$$E(p) = (\frac{\alpha^1}{\sum_{i=1}^K \alpha_i}, \frac{\alpha^1}{\sum_{i=2}^K \alpha_i}, \dots, \frac{\alpha^K}{\sum_{i=1}^K \alpha_i})$$

2.1.8 MCMC 和 Gibbs Sampling

在现实应用中，我们很多时候很难精确求出精确的概率分布，常常采用近似推断方法。近似推断方法大致可分为两大类：第一类是采样(Sampling)，通过使用随机化方法完成近似；第二类是使用确定性近似完成近似推断，典型代表为变分推断(variational inference)。

在很多任务中，我们关心某些概率分布并非因为对这些概率分布本身感兴趣，而是要基于他们计算某些期望，并且还可能进一步基于这些期望做出决策。采样法正式基于这个思路。具体来说，假定目标是计算函数 f(x)在概率密度函数 p(x)下的期望。

$$E_p|f| = \int f(x)p(x)\,dx$$

则可根据 p(x)抽取一组样本$x_1, x_2, \dots, x_N$，然后计算 f(x)在这些样本上的均值

$$\hat{f} = \frac{1}{N}\sum_{I=1}^{N} f(x_i)$$

以此来近似目标期望 E[f]。若样本$x_1, x_2, \dots, x_N$独立，基于大数定律，这种通过大量采样的办法就能获得较高的近似精度。可是，问题的关键是如何采样？对概率图模型来说，就是如何高效地基于图模型所描述的概率分布来获取样本。概率图模型中最常用的采样技术是马尔可夫链脸蒙特卡罗(Markov chain Monte Carlo，MCMC). 给定连续变量$x \in$ X的概率密度函数 p(x)，x 在区间 A 中的概率可计算为

$$\mathrm{P(A)} = \int_A p(x)dx$$

若有函数$f: X \to R$，则可计算 f(x)的期望

$$P(A) = E_p[f(X)] = \int_x f(x)p(x)dx$$

若 x 不是单变量而是一个高维多元变量 x，且服从一个非常复杂的分布，则对上式求积分通常很困难。为此，MCMC 先构造出服从 p 分布的独立同分布随机变量$x_1, x_2, \dots, x_N$，再

得到上式的无偏估计

$$\tilde{p}(f) = \frac{1}{N}\sum_{i=1}^{N} f(xi)$$

然而，若概率密度函数 p(x)很复杂，则构造服从 p 分布的独立同分布样本也很困难。MCMC 方法的关键在于通过构造“平稳分布为 p 的马尔可夫链”来产生样本：若马尔科夫链运行时间足够长，即收敛到平稳状态，则此时产出的样本 X 近似服从分布 p.如何判断马尔科夫链到达平稳状态呢？假定平稳马尔科夫链 T 的状态转移概率（即从状态 X 转移到状态X'的概率）为$T(x|x')$，t 时刻状态的分布为 p(x^t)，则若在某个时刻马尔科夫链满足平稳条件

$$p(x^t)T(x^{t-1}|x^t) = p(x^{t-1})T(x^t|x^{t-1})$$

则 p(x)是马尔科夫链的平稳分布，且马尔科夫链在满足该条件时已收敛到平稳条件。也就是说，MCMC 方法先设法构造一条马尔科夫链，使其收敛至平稳分布恰为待估计参数的后验分布，然后通过这条马尔科夫链来产生符合后验分布的样本，并基于这些样本来进行估计。这里马尔科夫链转移概率的构造至关重要，不同的构造方法将产生不同的 MCMC 算法。Metropolis-Hastings(简称 MH)算法是 MCMC 的重要代表。它基于“拒绝采样”（reject sampling）来逼近平稳分布 p。

吉布斯采样(Gibbs sampling)有时被视为 MH 算法的特例，它也使用马尔科夫链读取样本，而该马尔科夫链的平稳分布也是采用采样的目标分布 p(x).

2.2 LDA 模型

2.2.1 模型原理

在文本挖掘领域，大量的数据都是非结构化的，很难从信息中直接获取相关和期望的信息，一种文本挖掘的方法：主题模型（Topic Model）能够识别在文档里的主题，并且挖掘语料里隐藏信息，并且在主题聚合、从非结构化文本中提取信息、特征选择等场景有广泛的用途。主题模型有两种：PLSA（ProbabilisticLatent Semantic Analysis）和 LDA（Latent Dirichlet Allocation），本实验采用 LDA 模型即隐含狄利克雷分布模型进行建模。

LDA 是一种非监督机器学习技术，可以用来识别大规模文档集（document collection）或语料库（corpus）中潜藏的主题信息。它采用了词袋（bag of words）的方法，这种方法将每一篇文档视为一个词频向量，从而将文本信息转化为了易于建模的数字信息。

LDA 模型在结构上可以描述为一个由单词、主题、文档构成的三层贝叶斯网络[5]。模型的主要思想是将每篇文章看作所有主题的一个混合概率分布，而将其中的每个主题看作在单词上的一个概率分布。由此，当有 D 篇文档、T 个主题和 W 个单词时，在一篇文档中的第 i 个单词的概率可以表示为:

$$p(wi) = \sum_{j=1}^{r} p(wi|zi = j)p(zi = j)$$

在 LDA 模型中，参数 z 代表主题，参数 w 代表单词，则$p(zi = j)$表示的是从文档中取出一个单词属于主题 j 的概率，而$p(wi|zi = j)$代表的是当取出的单词属于主题 j 时该单词为 i 的概率。可以将$p(zi = j)$表示为文档在主题上的一个多项分布，记为$\theta_j = p(z = j)$,将$p(wi|zi = j)$表示为主题在单词上的一个多项分布，记为$\varphi_j w = p(w|z = j)$.

在以上生成式文档思想加入 Dirichlet 先验后，便得到人们熟知的 LDA 模型，其中θ表示文档在主题上的分布，φ表示主题在单词上的分布，再加入θ和φ的先验分布（分别服从参数α、β的 Dirichlet 分布），这样就能得到 LDA 模型的各层参数之间的依赖关系的数学表述：

$$wi|zi,\varphi^{(zi)}\sim Discrete(\varphi^{(zi)})$$
$$\varphi\sim Dirichlet(\beta)$$
$$zi|\theta^{(di)}\sim Discrete(\theta^{(di)})$$
$$\theta\sim Dirichlet(\alpha)$$

LDA 模型是一个概率生成式模型，其中一条文本生成的过程如下：①对于文档 d，从 Dirichlet (α) 抽样得θ；②对于主题 z，从 Dirichlet (β) 抽样得φ；③对于每个单词w_i及所属主题z_i，从多项式分布θ中抽样得$z_i = p(z_i|\theta)$,从多项式分布φ中抽样得$w_i = p(w_i|z_i,\varphi)$.

2.2.2 复杂度寻找最优主题数

在信息论中，perplexity（复杂度）用来度量一个概率分布或概率模型预测样本的好坏程度。它也可以用来比较两个概率分布或概率模型。低复杂度的概率分布模型或概率模型能更好地预测样本。在对文本的主题特征进行研究时，我们往往要指定 LDA 生成的主题的数目，而一般的解决方法是使用 perplexity 来计算，原理如下(概率分布 perplexity)[6]：

$$perplexity(D) = Exp\{\frac{-\sum_{d=1}^{M}\log(p(w_d))}{\sum_{d=1}^{M}N_d}\}$$

其中，M 是语料库的大小，N_d是第 d 篇文本大小（即单词个数）。

$$p(w) = \sum_{z} p(z)p(x|z,gamma)$$

其中 z 是主题，w 是文档，gamma 是训练集学出来的文本-主题分布。所以 perplexity 的上半部分就是生成整个文档的似然估计（表示训练出的参数的生成能力）的负值，由于概率取值范围为[0,1]，按照对数函数的定义，分子值是一个大数，而分母是整个语料集的单词数目。也就是说模型生成能力越强，perplexity 值越小。

为节约运算时间，本实验寻找主题数的步骤分为两步。首先，找出复杂度最低的主题数的大致范围。主题数选取范围为 20~100 个，步长为 5，进行模型训练，由图 7 可知，最佳主题数的范围大致在（30,40）之间。

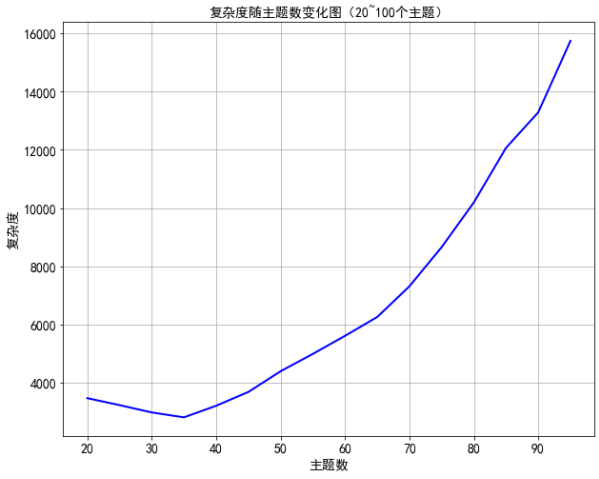

图 7 复杂度随主题数变化图（20~100 个主题）

其次，在该区间内再进行步长为 1 的详细划分，进行模型训练。最终，由图 8 复杂度随主题数变化图可知，本实验中当选取主题数为 35 时，复杂度最低。因此 35 应该为最优主题数。

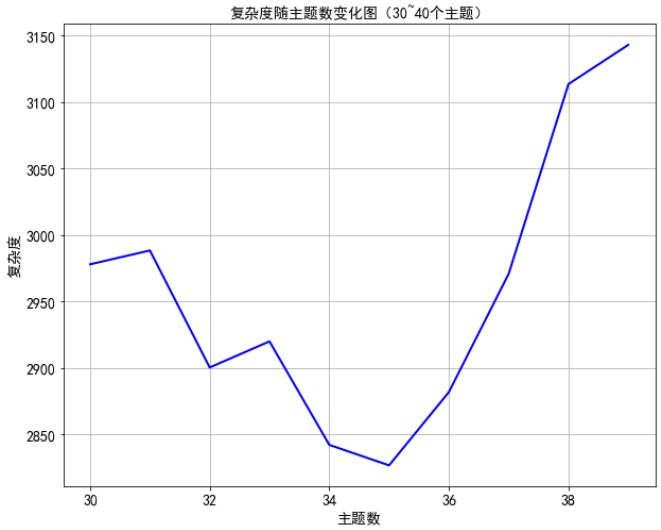

图 8 复杂度随主题数变化图（30~40 个主题）

2.3 构建热度指标

2.3.1 主题热度

本实验主题热度采用 LDA 模型学到的主题权重来代替。首先得到每个主题中各个词向量的权重值并将其求和代表该主题的主题权重。用着个主题权重值代表每个主题的主题热度。

2.3.2 外在热度

本文选取点赞数和反对数这两个数据来作为表征留言外在热度的参数。对于外在热度计算方式的思考来源于信息论中自信息量的描述：一个事件信息量的大小与该信息发生的概率有关，概率小的事件所包含的信息量大，概率大的事件包含的信息量小，则事件 A 的信息量的计算式为：

$$I(A) = -log\,(P(A))$$

类比于信息量的计算，假设某条留言 m 的点赞数为 c，反对数为 r，则此留言的外在热度的计算方式定义为：

$$outer\_heat(m) = -log_2 \frac{1}{c+1-r}$$

其中$\frac{1}{c+1-r}$可以理解为对该条留言 m 产生兴趣的人数有 c+1 人（包括留言者本人），r 代表不感兴趣或者不赞同本条留言的人数，应该减去。而在这个人群中，作为作者发表该留言的概率便是$\frac{1}{c+1-r}$。代入信息量的计算公式便得到这条留言的热度值。由于根据原始数据计

算会出现正无穷的现象，因此对于正无穷一概赋值为-1。

最后，计算根据 LDA 模型分配的主题的平均留言外在热度，即为主题 t 的外在热度。计算方式如下：

$$outer\_heat(t) = \frac{1}{N_t}\sum_{i=1}^{N_t} out\_heat(m_{ti})$$

其中，N_t表示属于第 t 个主题的留言数量，m_{ti}表示属于第 t 个主题的留言的外在热度。

2.3.3 总热度

总热度指标定义为主题权重（主题热度）和主题外在热度的加总，为了使两个指标在同一个数量级上，给主题的外在热度乘以 100，给主题权重除以 1000，以保持相同的数量级。

用所有留言的点赞数减去反对数代表外在热度的关注人数，计算结果为 12002.用留言的条数 4225 代表留言主题的关注人数。则主题热度和外在热度应分别赋予权重 1:2.81。

计算公式如下：

$$topic\_heat(t)' = \frac{1}{1000} topic\_heat(t)$$

$$outer\_heat(t)' = outer\_heat(t) \times 100$$

$$heat(t) = \frac{1}{2.81} topic\_heat(t)' + outer\_heat(t)'$$

2.4 实验

2.4.1 数据准备

本实验数据集采用第八届泰迪杯数据挖掘赛 C 题附件 3 提供的数据进行操作。导入数据后先进行去除空白噪声、去除重复留言等的预处理后下 4225 条投诉文本作为原始分析数据。

为了保证最后结果的可靠性和良好的可识别性，需要事先建立一个停用词表，该词表中包括常见的语气词、助词、无意义的尊称及感谢性话语等。使用该词表可以在分词的过程中便剔除掉大部分的噪声词语。

同时构建一个用户自定义词典，以保证一些固定地名和专有名词在 jieba 分词中不会被分开，来提高模型的准确性。

2.4.2 实验步骤

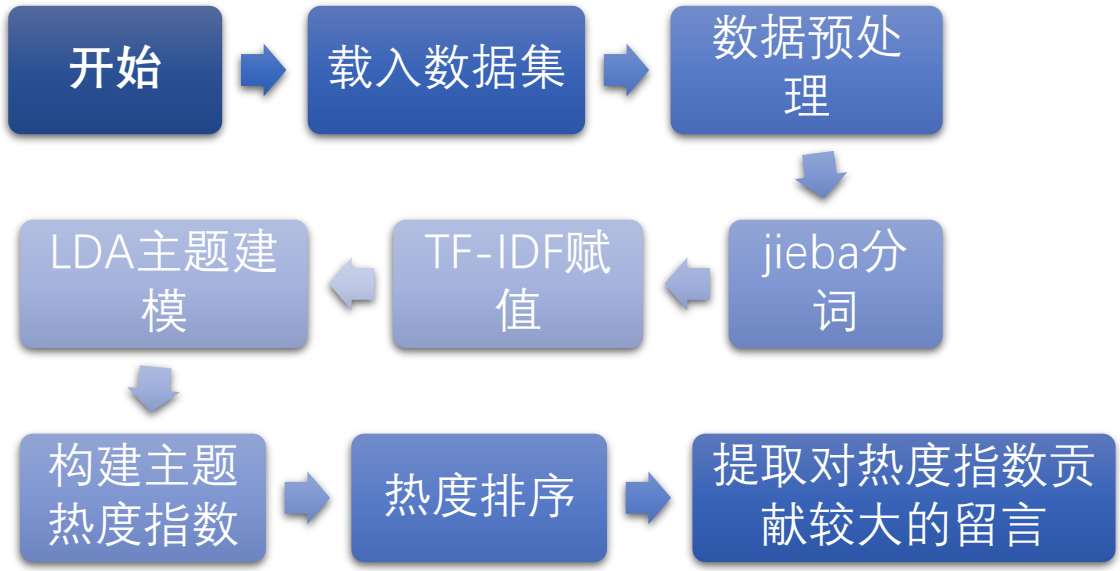

图 9 问题二实验步骤

2.4.3 实验结果及结论

本实验主要得出三个结果:
第一个结果为 LDA 模型的主题分布及各个主题中的热词, 并按照各个词的特征降序排列, 下表给出主题权重最大的前 5 个主题及它们最重要的前 10 个特征词:

Topic 20	Topic 5	Topic 14	Topic 31	Topic 8
• 红绿灯	• 市委	• 伊景园滨河	• 搅拌站	• 工地
• 路口	• 加快	• 捆绑	• 丽发新城小区	• 不买
• 直行	• 市政府	• 广铁集团	• 暮云街道	• 施工
• 车流量	• 背景	• 车位	• 扬尘	• 敏感
• 绿灯	• 倡导	• 定向	• 灰尘	• 通宵
• 3号	• 常年	• 铁路职工	• 粉尘	• 夜间
• 凉塘路	• 刻不容缓	• 销售	• 丽发新城	• 作业
• 地铁	• 落后	• 职工	• 搅拌	• 抢险
• 东四路	• 省会	• 购买	• 居民区	• 噪声污染
• 人流	• 近年	• 商品房	• 百米	• 建筑施工

图 10 主题权重最大的主题及它们的前 10 个特征词

观察主题 20 的前 10 个关键词，推断该主题似乎是关于交通设施的。再查看对这个主

题贡献最大的前五个留言，大致可以归纳出，主题 20 是“地铁站出口设置不合理问题”。

观察主题 5 的前 10 个关键词，推断该主题似乎是关于城市建设的。再查看对这个主题贡献最大的前五个留言，大致可以归纳出，主题 5 是“建议省市委加快城市建设步伐，改善城市落后面貌，给市民安居和谐的生活家园”。

观察主题 14 的前 10 个关键词，推断该主题似乎是关于小区车位配置。再查看对这个主题贡献最大的前五个留言，大致可以归纳出，主题 14 是“伊景园滨河小区捆绑销售车位问题”。

观察主题 31 的前 10 个关键词，推断该主题似乎是关于搅拌站污染问题。再查看对这个主题贡献最大的前五个留言，大致可以归纳出，主题 31 是“丽发新城小区附近搅拌站污染扰民问题”。

观察主题 8 的前 10 个关键词，推断该主题似乎是关于工地夜间施工问题。再查看对这个主题贡献最大的前五个留言，大致可以归纳出，主题 8 是“各地多处发生工地夜间施工扰民”。

第二个结果为，根据总热度降序排列的主题以及它们的主题热度、外在热度和总热度表。表 1 表 1 总热度最大的前 5 个主题给出总热度最大的前 5 个主题。

表 1 总热度最大的前 5 个主题

主题	排名	主题权重	外在热度	总热度
Topic 20	1	61.1334	80.7981	104.689
Topic 5	2	61.7432	81.8459	103.819
Topic 31	3	61.2036	73.3495	95.1302
Topic 2	4	50.4866	75.5419	93.5086
Topic 17	5	62.7417	65.9067	88.2347

第三个结果为各主题的留言详情。详细结果展示在实验附件中。

三、问题 3：答复意见的评定

3.1 同义词词林

《同义词词林》是梅家驹等人于 1983 年编纂而成，这本词典中不仅包括了一个词语的同义词，也包含了一定数量的同类词，即广义的相关词。由于《同类词词林》著作时间较为久远，且之后没有更新，所以哈尔滨工业大学信息检索实验室利用众多词语相关资源，完成了一部具有汉语大词表的“哈工大信息检索研究室同义词词林扩展版”。《同义词词林扩展版》收录词语近 7 万条，全部按意义进行编排，是一部同义类词典。哈工大信息检索研究室参照多部电子词典资源，并按照人民日报语料库中词语的出现频度，只保留频度不低于 3 的部分词语，剔除 14706 个罕用词和非常用词后，词表共包含 77343 条词语[7]。

同义词词林按照树状的层次结构(如图)把所有收录的词条组织到一起，把词汇分成大、中、小 3 类，大类有 12 个，中类有 97 个，小类有 1400 个。每个小类里都有很多的词，这些词又根据词义的远近和相关性分成了若干词群。每个词群中的词语又进一步分成了若干行，同一行的词语要么词义相同，要么词义有很强的相关性。

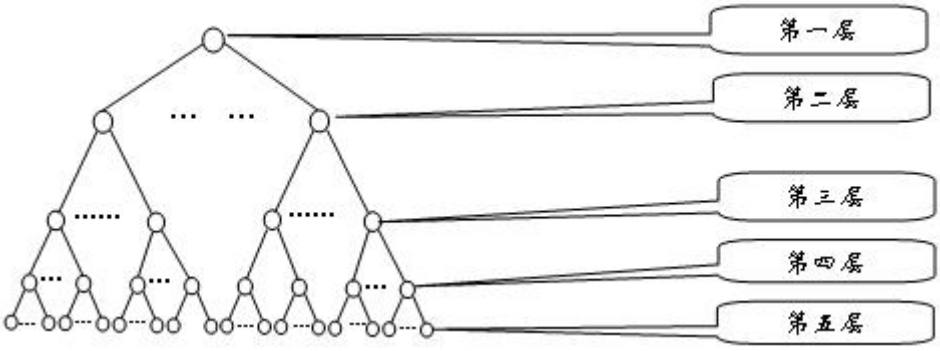

图 11 同义词词林

同义词词林提供了 5 层的树形编码，见下列具体文本，第 1 级用大写英文字母表示；第 2 级用小写英文字母表示；第 3 级用二位十进制整数表示；第 4 级用大写英文字母表示；第 5 级用二位十进制整数表示。例如：

```
Aa01A01= 人 士 人物 人士 人氏 人选
Aa01A02= 人类 生人 全人类
Aa01A03= 人手 人员 人口 人丁 口 食指
Aa01A04= 劳力 劳动力 工作者
Aa01A05= 匹夫 个人
Aa01A06= 家伙 东西 货色 厮 崽子 兔崽子 狗崽子 小子 杂种 畜生 混蛋 王八蛋 竖子 鼠辈 小崽子
Aa01A07= 者 手 匠 客 主 子 家 夫 翁 汉 员 分子 鬼 货 棍 徒
Aa01A08= 每人 各人 每位
Aa01A09= 该人 此人
Aa01B01= 人民 民 国民 公民 平民 黎民 庶 庶民 老百姓 苍生 生灵 生人 布衣 白丁 赤子 氓 群氓 黔首 黎民百姓 庶人 百姓 全民 全员
Aa01B02= 群众 大众 公众 民众 万众 众生 千夫
Aa01B03# 良民 顺民
Aa01B04# 遗民 贱民 流民 游民 顽民 刁民 愚民 不法分子 孑遗
Aa01C01= 众人 人人 人们
Aa01C02= 人丛 人群 人海 人流 人潮
Aa01C03= 大家 大伙儿 大家伙儿 大伙 一班人 众家 各户
Aa01C04= 们 辈 曹 等
Aa01C05@ 众学生
Aa01C06# 妇孺 父老兄弟 男女老少 男女老幼
Aa01C07# 党群 干群 军民 工农兵 劳资 主仆 宾主 僧俗 师徒 师生 师生员工 教职员工 群体 爱国志士 党外人士 民主人士 爱国人士 政
Aa01D01@ 角色
```

图 12 同义词词林示例

由于第 5 级有的行是同义词，有的行是相关词，有的行只有一个词，分类结果需要特别说明，可以分出具体的 3 种情况。使用特殊符号对 3 种情况进行区别对待，所以第 8 位的标记有 3 种，分别是“=”代表“相等”、“同义”；“#”代表“不等”、“同类”，属于相关词语；“@”代表“自我封闭”、“独立”，它在词典中既没有同义词，也没有相关词。

3.2 评价角度分析

3.2.1 相关性

相关性考察的是答复意见的内容是否与问题有关。

根据同义词词林编排特点，提出基于同义词词林的义项相似度，主要思想是基于同义词词林结构，利用词语中义项的编号，根据两个义项的语义距离，计算出义项相似度。在本题中，对留言详情和答复意见进行预处理，计算义项相似度，得出句子相似度作为相关性评价指标。义项相似度的算法如下：

首先判断在同义词词林中作为叶子节点的两个义项在哪一层分支，即两个义项的编号在哪一层不同。从第一层开始判断，相同则乘 1，否则在分支层乘以相应的系数，然后乘以调节参数 cos(n*pi/100)，其中 n 是分支层的节点总数，该调节参数的功能是把义项相似度控制在 0~1 之间。

词语所在树的密度，分支的多少直接影响到义项的相似度，密度较大的义项相似度的值相比密度小的相似度的值更精确。再乘以一个控制参数(n-k+1)/n，其中 k 是两个分支间的距离。

若两个义项的相似度用 sim 表示：

(1)若两个义项不在同一棵树上，sim(A,B)=f;

(2)若两个义项在同一棵树上：

①若在第 2 层分支，系数为 a:

$$sim(A,B) = 1 * a * \cos\left(n * \frac{\pi}{180}\right)\left(\frac{n-k+1}{n}\right)$$

②若在第 3 层分支，系数为 b:

$$sim(A,B) = 1 * 1 * b * \cos\left(n * \frac{\pi}{180}\right)\left(\frac{n-k+1}{n}\right)$$

③若在第 4 层分支，系数为 c:

$$sim(A,B) = 1 * 1 * 1 * c * \cos\left(n * \frac{\pi}{180}\right)\left(\frac{n-k+1}{n}\right)$$

④若在第 5 层分支，系数为 d:

$$sim(A,B) = 1 * 1 * 1 * 1 * d * \cos\left(n * \frac{\pi}{180}\right)\left(\frac{n-k+1}{n}\right)$$

在经过多次试验后，人工评定后将层数初值设置为 a=0.65,b=0.8,c=0.9,d=0.95。

3.2.2 可解释性

可解释性考察的是答复意见中内容的相关解释，可使用答复意见中包含留言主题共同关键词的占比来评定。

$$score = \frac{\sum 留言主题\ in\ 答复意见}{\sum 留言主题词语}$$

答复意见中的留言主题关键词越多，说明答复意见越围绕着留言主题来进行阐述，因而可解释性越强。

3.2.3 完整性

完整性考察的是答复意见是否满足规范。规范是指是否明确答复的对象（即留言用户），是否接收到相应的留言，是否进行答复，是否对留言表示感谢，是否标注答复时间。根据附件 4，可将以上五点规范明确为答复意见中是否包含“网友×× ： 您好”、“已收悉”、“回复如下”、“感谢您”、“××年××月××日”。采用十分制，每点 2 分。若大于 6 分，则答复意见评定为“十分完整”；若为 6 分，则答复意见评定为“完整”；若小于 6 分，则答复意见评定为“不完整”。

3.3 实验

3.3.1 数据准备

本实验数据集采用第八届泰迪杯数据挖掘赛 C 题附件 4 提供的数据进行操作。先自行创建一个 load_cilin 模块，定义好相似度、相关性、可解释性以及完整性的函数，在后续的实验操作中直接导入这个模块进行建模。

3.3.2 实验步骤

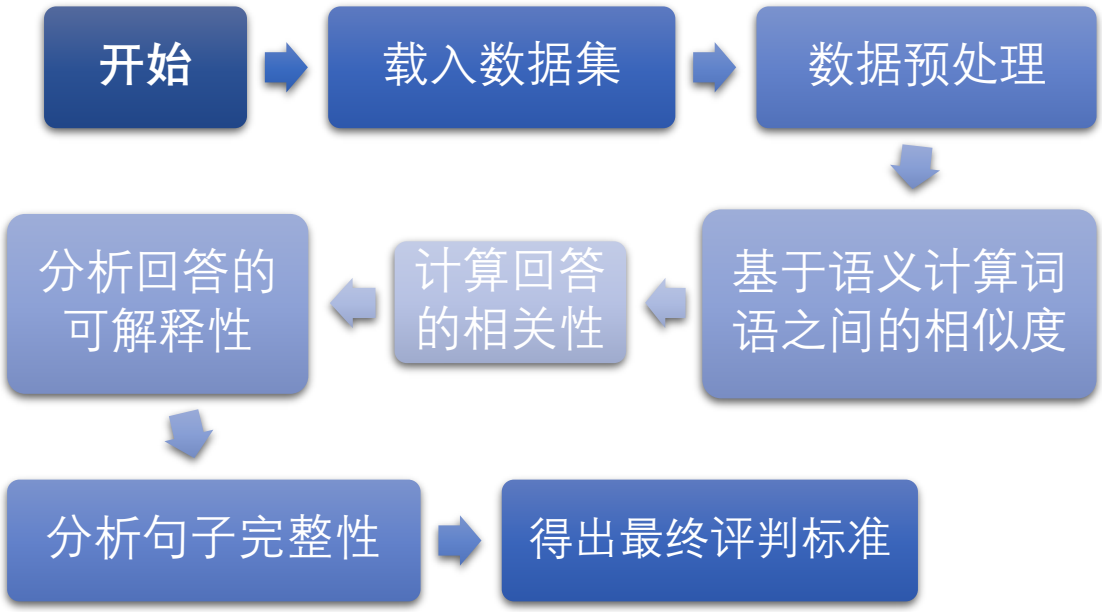

图 13 问题三实验步骤

3.3.3 实验结果

实验获得关于回复的相关性、完整性和可解释性三个结果，表 2 给出前五个留言回复的评价方案效果:

表 2 前五个留言回复的评价方案

留言编号	留言主题	相关性	完整性	可解释性
2549	A2 区景蓉华苑物业管理有问题	71.95	十分完整	38.90
2554	A3 区潇楚南路洋湖段怎么还没修好?	71.15	十分完整	60
2555	请加快提高 A 市民营幼儿园老师的待遇	80.49	十分完整	88.89
2557	在 A 市买公寓能享受人才新政购房补贴吗?	82.96	十分完整	75.45
2574	关于 A 市公交站点名称变更的建议	71.43	十分完整	67.14
. . .	. . .	. . .	. . .	. . .

以前两条留言的评价体系为例。留言答复 2549 与留言答复 2554 相比，两者答复意见

的相关性和完整性均较为相似，但是以可解释性而言，留言答复 2549 明显低于留言答复 2554。也就是说，相较于留言答复 2554，留言答复 2549 不能视为较好地对用户留言进行解释。完整评价结果展示在实验附件中。

参考文献

[1]李清镇. 基于文本挖掘的笔记本电脑网评分析[D].兰州财经大学,2019.
[2]王博. 文本分类中特征选择技术的研究[D].国防科学技术大学,2009.
[3]吴萍萍.基于信息熵加权的 Word2vec 中文文本分类研究[J].长春师范大学学报,2020,39(02):28-33.
[4]玉龙.隐含狄利克雷分布[EB/OL].https://www.zybuluo.com/learning17/note/1167651,2018.
[5] BLEI D M,NG A Y,JORDAN M I.Latent dirichlet allocation[J].Journal of Machine Learning Research,2003,3:993-1022.
[6]王婷婷, 韩满, 王宇.LDA 模型的优化及其主题数量选择研究——以科技文献为例[J].数据分析与知识发现.2018, 13 (1)
[7]田久乐,赵蔚.基于同义词词林的词语相似度计算方法[J].吉林大学学报(信息科学版),2010,28(06):602-608.