“智慧政务”中的文本挖掘应用

摘 要：随着网络问政平台的逐步普及，群众向政府反映问题的渠道愈加便捷，，这也导致各类社情民意相关的文本数据量与日俱增，给以往主要依靠人工来进行留言划分和热点整理的相关部门的工作带来了极大挑战。为了缓解当前政务系统的压力、提高政府施政效率，本文通过多种自然语言处理方法，针对附件中给出的来自互联网公开来源的群众问政留言内容，展开了文本分类、热点问题挖掘等任务。

针对附件中给出群众留言内容和相关部门答复意见的预处理。包含中文分词、停用词去除、词性标注、单字词去除等预处理步骤。此外，由于需要训练词向量模型，针对其来自维基百科中文语料库中的数据还进行了繁体转简体的处理步骤。而为了提高最终词向量模型的效果，还将此次附件中的所有中文内容都追加到用于训练词向量模型的维基百科中文语料库中。

针对任务一，本文建立一种关于留言内容的一级标签分类模型，在将留言内容构建成文本向量空间后，通过训练出的支持向量机文本分类器用以分类留言内容。不过在构建文本向量空间模型的过程中，相较于传统方法中用 TFIDF 提取词汇特征，本文提出了一种融合词频、词性、词位置特征的词汇特征提取方法，在减小词袋长度的同时，也取得了较好的文本分类效果。

针对任务二，本文通过 LDA 主题模型来获取留言内容的主题详情，通过计算每条留言与不同主题之间的关联程度，判断每条留言的所属主题。而后，通过训练出的 FastText 词向量模型，参照均值词向量构建文本向量的思路，计算出每个主题的主题向量，在合并相似主题的同时实现对文本的进一步归类。此外，还提出一种热度评估方法来获取热度靠前的五个热点问题。

针对任务三，本文创建一套相关部门对留言的答复意见的评价方案。本文从答复与留言之间的相关性、答复对于留言的完整性以及答复的及时性三个方面提出了一种答复意见的质量评价方法。

在论文的最后，浅谈了此次实验的感想和收获。

关键词：文本分类；支持向量机；热点挖掘；LDA 主题模型；FastText 模型

Text Mining Application in "Smart Government Affairs"

Abstract: With the gradual popularization of online questioning platforms, people can reflect problems to the government through more and more channels, which also leads to the increasing amount of text data related to various social conditions and public opinion. At the same time, it also brought great challenges to the work of the relevant departments that used to manually divide messages and organize hotspots. In order to alleviate the pressure of the current government affairs system and improve the government's governance efficiency, this article uses a variety of natural language processing methods to expand the tasks of text classification and hot topic mining for the content of public questioning messages from public sources given in the annex.

For the pre-processing of the mass message content given in the attachment and the responses from relevant government department. it includes pre-processing steps such as Chinese word segmentation, stop word removal, part-of-speech tagging, and single-word word removal. In addition, due to the need to train the word vector model, the data from Wikipedia Chinese corpus has also been processed from traditional to simplified. In order to improve the effect of the final word vector model, all the Chinese content in this attachment is also added to the Wikipedia Chinese corpus for training the word vector model.

For the task one, this paper establishes a first-level label classification model about the content of the message. After constructing the message content into a text vector space, the trained support vector machine text classifier is used to classify the message content. However, in the process of constructing the text vector space model, compared with the traditional method of using TFIDF to extract vocabulary features, this paper proposes a vocabulary feature extraction method that combines word frequency, part of speech, and word position features, while reducing the length of the word bag, also achieved better text classification results.

For the task two, this paper uses the LDA topic model to obtain the topic details

Keywords: text classification; support vector machine; hot spot mining; LDA topic model; FastText mode

目录

一、引言

伴随着“互联网+政务”的新型模式在各级政府进一步推广与实施，相关社情民意政务数据正呈指数级的速度快速增长。以往依靠人工进行社情民意类别划分和热点整理的政务处理模式，已逐渐不能满足当下群众的问政需求。如何使现有政务系统进一步智慧化和高效化，正在成为一个迫在眉睫的问题。

自然语言处理是近些年人工智能领域的热点话题之一。随着大数据、云计算等技术的迅猛发展，许多曾经认为需要强大算力才能处理的文本挖掘任务成为了可实现的目标。文本挖掘任务中最常见的两项工作便是文本分类和聚类。对于文本分类任务，我们可以根据预先给定文本类别的数据所训练出的文本分类器，对不同内容形式的文本进行自动分类，从而摆脱人工分类的低效窘况，提升工作效率。而对于文本聚类任务，我们可以根据不同的需求将文本内容按照一定的相似性准则进行自动归类整合，以此解决在没有类别标签的数据中的文本归类问题。

因此，针对此次“智慧政务”中的文本分类、聚类问题，我们希望能够通过相关自然语言处理技术，更好的实现对各类社情民意相关文本的挖掘任务，以此来促进当前政务处理朝智慧化和高效化方向的发展。

二、问题分析

2.1 群众留言分类问题分析

问题一要求，根据附件 1 给出的留言内容划分体系，构建一个关于附件 2 中留言内容的一级标签分类模型，且该模型能够在 F-Score、查准率等评价指标上取得较好的结果。

针对此问题，我们对比了以往研究中所采用的多种文本分类模型，综合分析后决定采用支持向量机分类器来实现对附件 2 中留言内容的分类。同时，为了有效利用附件 2 中所给出的数据，我们将留言主题与留言详情合并，构建了一个新的标签“留言主题与详情”，并以此标签下的数据作为此次实验的数据。

2.2 留言内容热点挖掘分析

问题二要求，根据附件 3 中给出的留言内容进行有效的归类，并定义出一个

合理的热度评价指标，在准确挖掘出留言主题中的热点问题的同时，还能依据自定义的热度评价指标对各个留言主题的热点问题进行热度分析，以此挖掘出最受关注的热点问题都有哪些。

针对这些问题，我们决定采用 LDA 主题模型实现对留言内容的主题挖掘，此外，根据附件 3 中给出的数据标签，我们结合了点赞数量、反对数量、同一主题留言数量多个影响因素来构建热度评价指标，以此实现对热点问题的挖掘和热点评价。

2.3 答复意见质量评价分析

问题三要求，根据附件 4 给出的相关部门对留言的答复意见，从相关性、完整性等角度为每条答复意见构建一个质量评估体系，来衡量相关部门针对留言的回复意见的质量。

针对此问题，我们构建了一个以答复意见和留言内容的相似性、答复意见的完整性、答复时间对留言时间的及时性为评价标准的质量评估体系，从三方面对答复意见的质量进行分析，并整合出一份答复意见质量评分表，以此来衡量答复意见的质量。

三、 数据预处理

3.1 标签合并

此次数据集中给出的数据标签包含留言用户、留言主题、留言时间、留言详情等等，其中，留言主题是对留言详情的高度概括，为了充分利用实验数据，将留言主题的内容与对应的留言详情的内容进行合并，构建一个新的标签“留言主题与详情”。以下文中所提及的“留言内容”均指该新创建的标签内容。

3.2 中文字符提取

综合分析留言数据后，发现留言内容中所包含的数字字符和英文字符往往对留言主题的贡献度较低，如“A 市”、“A3 区”等，此次留言数据分析仅考虑中文部分的内容，通过正则表达式"[\u4e00-\u9fa5]"提取相应标签中的中文字符，过滤英文字符和数字字符。

3.3 分词

不同于英文文本中词与词之间用空格进行分隔，在中文文本里，词与词之间没有明显的界限用来分隔，因此从中文文本中提取词语时需要进行中文分词。本文通过 python 内置的分词模块 jieba 分词来进行中文分词。

3.4 词性标注

文本中不同词性对于文本主题的贡献程度往往有所区别，一般认为动词、名词和形容词对于文本主题贡献度较大，其他词性的主题贡献度则相对较差一些。此次实验通过 python 的 jieba 分词对文本中词语的词性进行标注。

3.5 停用词去除

停用词是指那些中文本中出现频繁却没有什么实际含义的词汇，在中文文本中，诸如“的”、“是”、“各自”、“同时”等等都是停用词，这些停用词在文本分类、聚类等实际应用中往往会对最终结果产生较大的干扰，因此需要剔除。现有中文停用词剔除的方法一般是通过和停用词表对照的方法来去除停用词，此次实验中将百度停用词表，哈工大停用词表，四川机器实验室停用词表合并成一个停用词表以提高停用词去除效果。

3.6 单字词去除

经过上述预处理步骤后，文本中仍然存在大量单字词。中文中，往往由两个字以上构成的词汇，能够更好的表现词义，而单字词的词汇表现力往往较差。为减少干扰，此次实验不考虑单字词。

3.7 繁体转简体

此次实验中涉及词向量的训练。词向量的训练语料是从维基百科中下载的中文语料库，该语料库中包含大量繁体字，为方便后期实验，通过 python 的 zhconv 库将下载的维基百科语料库中包含的繁体字全部转为简体字。

3.8 词向量训练语料库内容追加

题目给出的数据中包含中文内容的标签主要是留言主题、留言详情和答复意见，为了提高词向量训练的准确率，我们将上述标签中的中文数据全部提取出来追加到维基百科中文语料库中，用以作为词向量模型的训练语料库。

四、 构建留言内容的一级分类模型

4.1 留言内容分类方案介绍

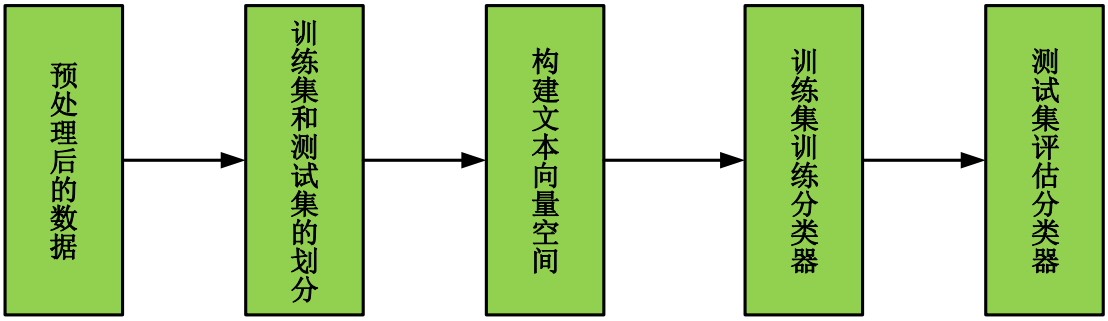

图 1. 模型构建流程图

如上图流程图所示，此次实验中，首先将数据集划分为训练集和测试集，训练集和测试集的比例是 3:1；而后，利用本文提出的特征融合的方法构建文本向量空间，并用支持向量机作为此次留言分类的分类器。此外，为了保证实验结果更加贴近实际结果，实验过程中分别进行了 10 次数据集的划分，并将每次划分的训练集和测试集进行实验，取 10 次实验结果的均值作为最终分类器的分类效果。

4.1.1 训练集和测试集的划分

为了训练出合理有效的留言内容分类器，需要将留言数据集进行划分，即从每个类别中随机选取 75%的留言内容作为训练文本集，剩余的 20%作为测试文本集。

4.1.2 文本向量空间构建

向量空间模型(Vector Space Model，VSM)，是目前应用最为广泛的文本表示模型[1]。该模型将文本中的词语映射到向量空间中，形成文本中文字和向量数据之间的映射关系。文本的每个特征词对应文本向量的每一维，特征词所对应的值

为该特征在文本中所占的权重值。

该模型的数学表述如下：文本数据集D $= (d_1, d_2, \dots, d_m)$，m表述数据集 D 中所包含的文本数量；词汇特征集$T = (t_1, t_2, \dots, t_n)$，$n$表示数据集 D 中所包含的词袋长度；任意一篇文本都可以表示为$V(d_i) = \{t_1, w_{i,1}; t_2, w_{i,2}; \dots; t_n, w_{i,n}\}$，$w_{i,n}$ 表示第i篇文本中第n个特征项的权重值。

向量空间模型的优点是把对文本的处理转化成对向量的处理，极大的简化了问题的复杂度、提高了文本的处理速度，但是该模型忽略词汇可能存在的一词多义、近义词等现象，同时在处理海量文本数据是，还会导致文本特征向量维度过度且过于稀疏的问题、为最终计算带来较大的负面影响。

构建文本向量空间模型时量化文本词汇特征最常用的方法之一，而目前应用最广泛的词汇特征提取方法是基于词频-逆向文本频率(TF-IDF)的方法。但是，TFIDF 值实际上是一种词频特征，用此特征来构建文本向量空间模型时往往会忽略词汇的位置特征、词性特征等等。在文本中，词汇所在不同位置往往意味着其重要性的区别，出现在标题中的词语往往会比出现在正文中的词汇更重要。而对于词性来说，大量研究表明动词、名词和形容词往往会比其他词汇具有更好的主题表现力。故这里提出一种多特征融合的词汇特征提取方法，用以替换传统的 tfidf 的算法。

1. 词频特征

此次采用的词频特征是词频-逆文档频率（TFIDF）。TFIDF 是描述词汇在文本中重要性最常用的特征之一，它的主要思想是若某个词语在某篇文章中出现的次数较多、而在其他文章中则很少出现，则认为这样的词语具有较好的类别区分能力[2]。TFIDF 实际上是 TF 词频(Term Frequency)与 IDF 逆文档频率(Inverse Document Frequency)乘积的结果，具体公式如下：

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

其中$n_{i,j}$是词语t_i在文本d_j中出现的次数，$\sum_k n_{k,j}$表示在文本d_j中 k 个总词语的次数之和。

$$idf_i = \log \frac{|D|}{|\{j: t_i \in d_j\}|}$$

其中$|D|$表示总文本数量，$|\{j: t_i \in d_j\}|$表示包含词语t_i的文本数量。

2. 词性特征

许多学者的研究都表明，动词、名词和形容词对于文本主题的表现力往往比

其他词性的词汇好一些，因此在经过词性标注后，给不同的词汇根据其词性赋予不同的权重，此次词性权重的设置参照文献[4]中提到的方法，分别设置为：

$$pos(\omega_{m,n}) = \begin{cases} 0.8 & if\ \omega_{m,n}\ is\ noun \\ 0.5 & if\ \omega_{m,n}\ is\ verb \\ 0.4 & if\ \omega_{m,n}\ is\ adj \\ 0.1 & if\ \omega_{m,n}\ is\ others \end{cases}$$

3. 位置特征

在标题中出现的词汇往往更能体现词语主题，这里将留言主题标签视为标题，参照文献[4]中词汇标题权重的设置方法，若词汇在留言主题中出现，则设置为 1，反之设置为 0。

综合上述特征，提出一种特征融合的词权计算公式：

$$\text{weight} = \alpha * \text{tfidf}_{\text{feature}} + \beta * \text{position\_feature} + \gamma * \text{pos\_feature}$$

其中，α表示词频特征所占权重比例，β表示位置特征所占权重比例，γ词性特征所占权重比例，且满足$\alpha + \beta + \gamma = 1$。

在构建文本向量空间之前，首先用上述公式计算每条留言中各个词汇的词权，提取词权靠前百分之八十的词汇用于表征整条留言，这样可极大的减小词袋的长度、滤除不必要的干扰词汇。而后，用提取出的百分之八十的词汇用来构建文本向量空间，并将文本向量空间中每个词的权重用上述公式求出的权重来表示，这与传统方法中仅用 tfidf 值来表示词汇特征不同，这里综合了 tfidf 值、词语位置权重和词性权重来表征词汇权重。在参数的设置上，参考网格搜索的思路，在计算词权用以提取百分之八十权重靠前的词汇时，设置α为 0.4，β为 0.1，γ为 0.5；在构建文本向量空间的过程中，设置α为 0.8，β为 0.1，γ为 0.1。

综合来看，这种方法既减少了文本向量空间的维度，又用一种新的词汇特征提取方法来提取文本词汇特征。

4.1.3 基于支持向量机的文本分类器

支持向量机(Support Vector Machine)是 Cortes 和 Vapnik 于 1995 年首先提出的[5]，它的重要理论基础是统计学习理论的 VC 维理论和结构风险最小化原理，能较好的解决小样本学习、非线性以及高维度识别问题，是当前机器学习领域特别是文本分类领域的研究热点之一。

所谓 VC 维是对函数类的一种度量，可以简单的理解为问题的复杂程度，VC 维越高，一个问题就越复杂。正是因为 SVM 关注的是 VC 维，SVM 解决问题的时候，和样本的维数是无关的，这使得 SVM 很适合用来解决文本分类的问题。

相对于其他所有文本分类方法，支持向量机具有更好的分类性能：

(1)文本向量维数很高，对于高维问题，支持向量机具有其它机器学习方法不

可比拟的优势；

(2)文本向量特征相关性大，许多文本分类算法建立在特征独立性假设基础上，受特征相关性的影响较大，而支持向量机对于特征相关性不敏感；

(3)文本向量存在高维稀疏问题，一些文本分类算法不同时适合于稠密特征矢量与稀疏特征矢量的情况，但支持向量机对此不敏感；

(4)文本分类样本收集困难、内容变化迅速，支持向量机能够找出包含重要分类信息的支持向量， 是强有力的增量学习和主动学习工具，在文本分类中具有很大的应用潜力。

4.1.4 测试集评估分类器效果

为了使实验结果更加贴近实际值，实验中进行了 10 次文本数据训练集和测试集的划分，每次都是在基于不同随机种子的前提下划分数据，然后分别对每次数据划分的结果进行分类器训练和评估，取 10 次实验的结果的平均值作为最终结果。

此次采用精确率、召回率和 F 值三个指标对文本分类器的分类效果进行评估。

精确率（precision）用来表示预测出来的正例中有多少是真的正例

$$\text{Precision} = \frac{TP}{TP + FP}$$

召回率（recall），描述的是所有正例能被发现多少

$$\text{Recall} = \frac{TP}{TP + FN}$$

F 值是精确率和召回率的调和均值

$$\text{F1} - \text{Score} = \frac{2PR}{P + R}$$

其中，TP 表示正例预测正确的个数；FP 表示负例预测错误的个数；TN 表示负例预测正确的个数；FN 表示正例预测错误的个数。

4.2 实验结果分析

4.2.1 准确率、召回率和 F 值

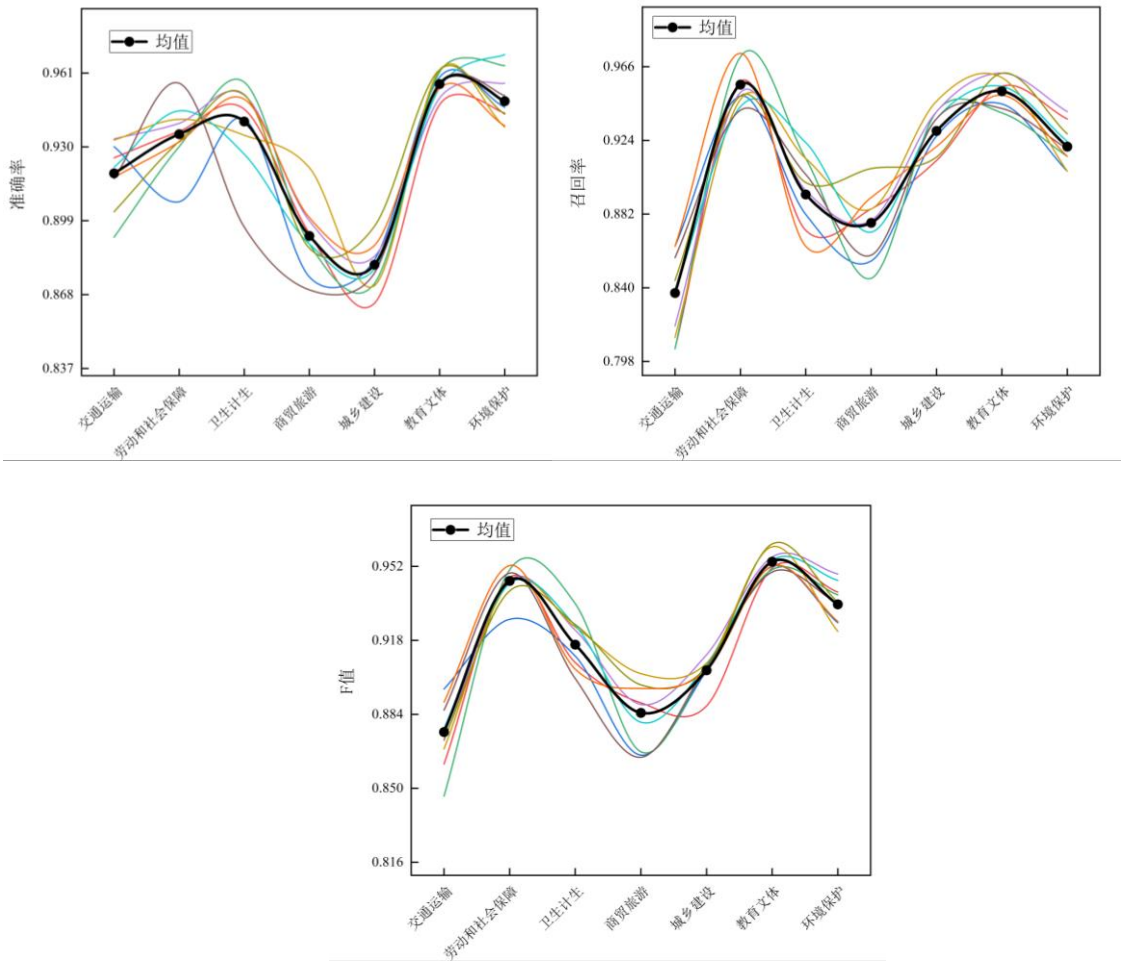

图 2. 混淆矩阵评价结果

为了使实验结果更加贴近实际值，分别进行了十次实验，并求取每次文本分类实验结果的精确率、召回率和 F 值。如上图所示，彩色曲线分别表示每次的三种评价指标每次的实验结果，黑色曲线表示三种评价指标各自十次实验的均值。以均值作为最终文本分类的结果，求得精确率达 0.924794651，召回率达 0.909258，F 值达 0.916398798。十次实验的结果展示在附录中。

4.2.2 混淆矩阵绘制的热力图展示

分别进行了 10 次实验，每次实验都会产生一个混淆矩阵，通过混淆矩阵可以直观地看出不同类别留言的分类结果。这里将 10 次实验所产生的 10 个混淆矩阵求和后再取均值，以此作为最终的混淆矩阵结果，如下图所示：

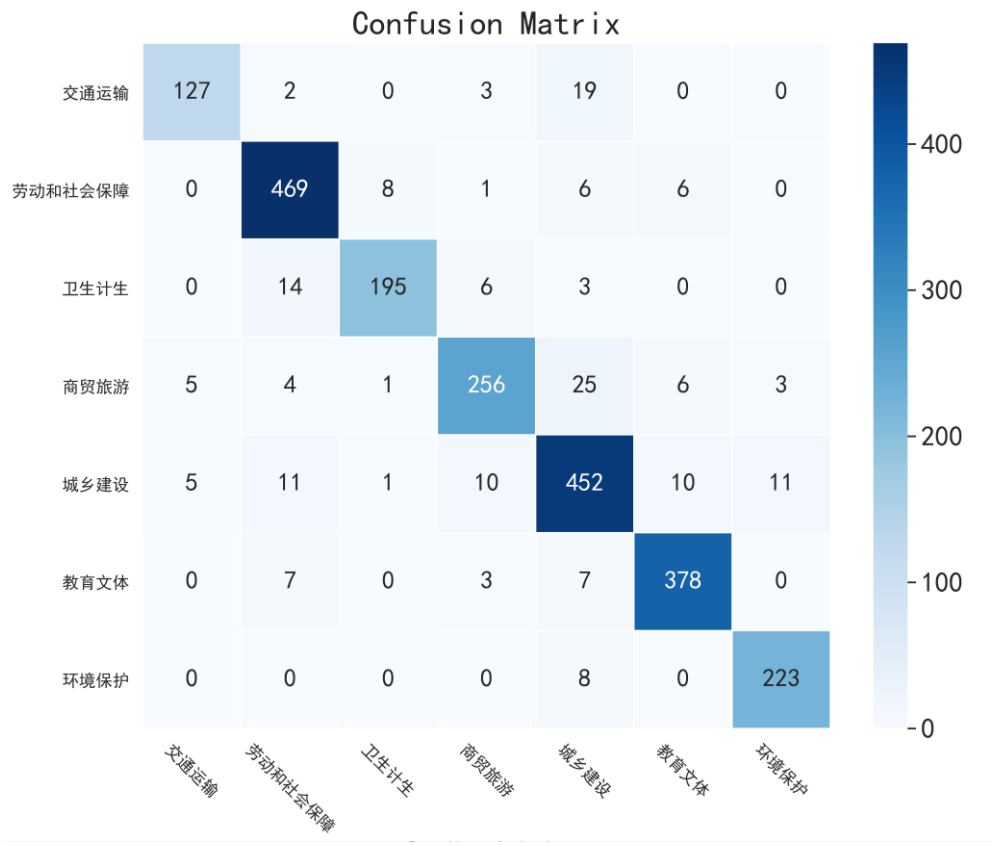

图 3. 热力图结果

可以看到，教育文体类别和环境保护类别中的留言分类效果都比较好，而城乡建设类别的留言中被误分类的数量是最多的；同时，商贸旅游类别和交通运输类别被误分类的留言里，有很大比重都是和城乡建设类别有关。卫生计生类别中有较多的误分类结果与劳动和社会保障类别有关。劳动和社会保障类别中的留言数量是最多的，尽管也有不少误分类结果，但总体效果都比较好。

五、热点问题挖掘

5.1 留言内容热点挖掘方案介绍

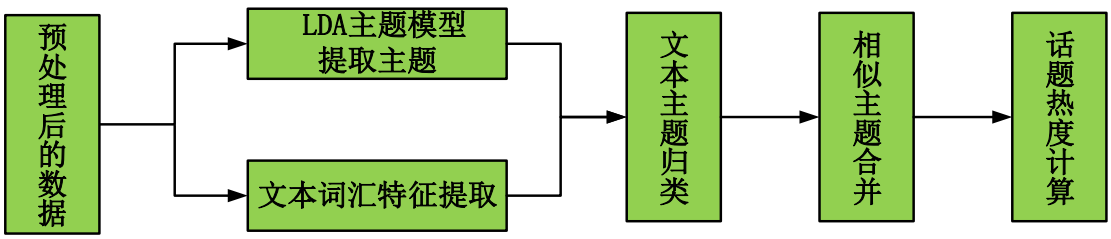

图 4. 热点问题挖掘流程图

如上图流程图所示，首先，本实验通过 LDA 主题模型实现对留言内容的主题建模，而作为 LDA 主题模型中最重要参数之一的主题数量，则是通过困惑度值来确定。其次，提出了一种多特征融合的词语权重计算方法，用以衡量留言中每个词汇的重要性。第三，判断每一篇留言与各个主题之间的相似程度，将每篇留言与其最相似的主题归类。第四，计算 LDA 主题模型提取出的各个主题之间的相似度，合并相似主题以减少主题数量，同时合并相似主题中所包含的留言内容。最后，结合点赞数、反对数、同一主题中留言数量三个指标构建主题热度评估函数，为每个主题计算热度值。

步骤四中提到的计算主题相似度，是利用均值词向量来构建文本向量的方法，构建主题向量后再通过余弦相似度原理合并相似主题。

主题向量的构建需要借助词向量模型来实现，此次采用 FastText 词向量模型来构建主题向量。以下对 FastText 词向量模型作简单介绍。

5.1.1 Fasttext 词向量模型

FastText[6][7]方法包含 3 部分：模型架构、层次 softmax 和 N-gram 特征。

FastText 模型输入一个词的序列（一段文本或者一句话），输出这个词序列属于不同类别的概率。序列中的词和词组组成特征向量，特征向量通过线性变换映射到中间层，中间层再映射到标签。FastText 在预测标签时使用了非线性激活函数，FastText 模型架构和 Word2Vec 中的 CBOW 模型很类似。不同之处在于，FastText 预测标签，而 CBOW 模型预测中间词。

在某些文本分类任务中类别很多，计算线性分类器的复杂度高。为了改善运行时间，FastText 模型使用了层次 Softmax 技巧。层次 Softmax 技巧建立在哈弗曼编码的基础上，对标签进行编码，能够极大地缩小模型预测目标的数量。

FastText 模型对输入的词序列加入了 N-gram 处理，用来解决词顺序丢失的问题。具体做法是把 N-gram 当成一个词，用 embedding 向量表示，在计算隐层时，把 N-gram 的 embedding 向量也加进去求和取平均。

此次实验通过维基百科中文语料库中的数据来训练 FastText 词向量模型，此外，为了提高最终词向量模型的效果，还将此次附件中的所有中文内容都追加到用于训练词向量模型的维基百科中文语料库中，并设置词向量维度为 300 维。

5.1.2 困惑度判断主题数量

困惑度值是评估语言模型性能的常用指标，其基本思想是给测试集赋予较高概率值的语言模型较好。一般来说，困惑度值会随着主题数量的增加而呈现递减

趋势。

困惑度值的计算公式如下：

$$\text{perplexity}(D) = \exp\left\{\frac{\sum_{d=1}^{M}\log p(w_d)}{\sum_{d=1}^{M}N_d}\right\}$$

其中，D表语料库中的文档，共有M篇，N_d表示每篇文档d中的单词数，w_d表示文档d中的词，$p(w_d)$即文档中词w_d产生的概率。

主题数量是主题模型中最重要的参数之一。近年来，有许多学者通过困惑度值来判断主题模型的主题数量。

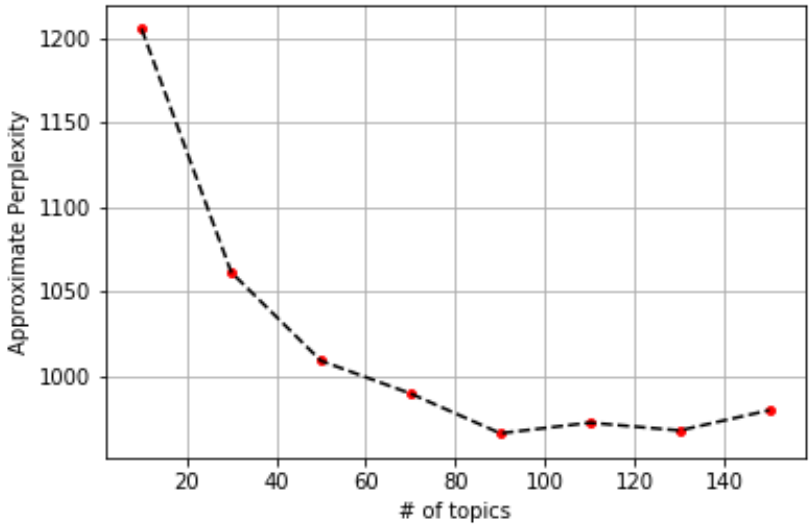

图 5. 困惑度值曲线

通常来说，困惑度值越小，说明对于文档的不确定性越小，模型的推广性越好。如图 2 所示，随着主题数的增加，困惑度值呈现递减规律。当主题数目取 90 时困惑度值最小，因此本文设定主题数量为 90。

值得一提的是，尽管有大量的研究是通过困惑度值来判断主题数量，但是该指标也存在着一些缺陷，即用困惑度值判断主题数量常常会导致主题数量过多、主题之间辨析度不明显的问题[8]，因此实验中通过求出主题向量、计算主题之间的余弦相似度来合并相似主题，详细的将在 5.1.6 章节中介绍。

5.1.3 LDA 主题模型提取主题词

LDA 主题是一种挖掘文本潜在主题的概率生成模型，由 Blei 等(2003)首次提出[9]，它是一种三层贝叶斯模型，分为文档层，主题层及主题关键词层，每层均有相应的随机变量或参数控制。该模型认为每篇文档都由一系列主题按照一定的概率混合形成，而每个主题又是由一系列单词按照一定的概率混合形成。图 7 展示了 LDA 主题模型原理图。

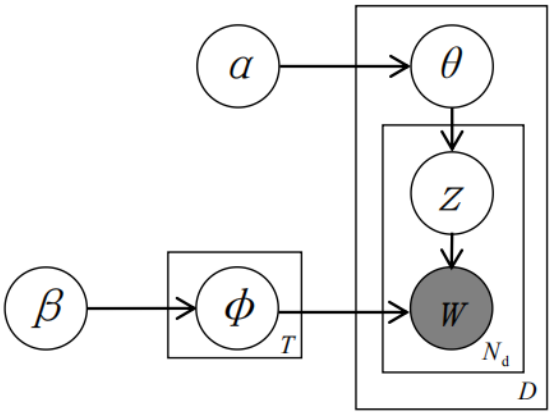

图 6. LDA 主题模型原理图

其中D表示数据库内文档总数量，T表示主题数量，θ表示文档的主题分布，φ表示主题的单词分布，α和β分别是θ和φ的狄利克雷超参数，z表示主题，W则表示属于主题z的单词。

LDA 主题模型的生成文档的过程可概括为以下三个步骤：

1. 从文档D对应的多项分布θ中提取每个词对应的主题 z。
2. 从主题z内提取单词W，同时该主题对应的多项分布为φ。
3. 重复上述(1)(2)两个步骤，共计重复N_d次，直到将文档内的所有单词都遍历一遍。

本研究使用了 python 编程语言中的 sklearn 库内的 LatentDirichletAllocation 函数来实现对于留言内容的主题建模。用 LDA 主题模型进行主题挖掘时，设置主题数量 K 为 90，超参数 α 为 50/k，β 为 0.01[9]，主题挖掘部分结果如表 1 所示：

表 1. 主题挖掘部分结果

主 题	主 题 关 键 词
主题 5	办理 社保 证明 网上 系统 窗口 记录 客服 业务 查询 身份证 工作人员 办事 生育 资料
主题 21	车辆 停车 停车场 停车位 交警 现象 占用 违停 停放 路边 道路 车道 交通事故 乱放 车子
主题 32	装修 价格 精装 造价 消费者 公示 楼盘 销售 提供 政府 计算 精装修 第三方 核算 清单
主题 44	经营 油烟 营业 门面 扰民 管道 楼下 东路 夜宵 烧烤 外面 深夜 魅力 管管 临街
主题 80	施工 扰民 噪音 凌晨 夜间 工地 作业 城管 噪声 安静 连续 中海 半夜 持续 通宵

5.1.4 文本词汇权重计算

对任意一条留言内容，其中所包含的词汇对于整条留言想反映的主题的贡献程度有所区别的[10]。分析热点话题的过程中，需要充分利用那些对于主题贡献度较大词汇。这里利用 5.1.2 提出的多特征融合的词权计算方法来量化每个词在留言中的重要性，在参数的设置上与 5.1.2 保持一致，设置 α 为 0.5， β 为 0.4， γ 为 0.1。

5.1.5 基于主题的留言归类

在 LDA 主题建模时，从每个主题中提取四十个主题关键词，同时计算每条留言中所包含词汇的权重值。对任意一条留言，分别与每个主题的对应主题关键词取交集，并选择交集最大、即重叠词汇最多的的主题为该留言所属主题。如表 2 所示：

表 2. 部分留言归类结果

留言编号	留言内容	主题编号	重叠主题	重叠词汇
285560	社保 卡制卡 进度 未知 制卡 进度 配送 阶段 归 市 银行 咨询 社保卡 服 务 窗口 银行 制卡 流程 说法不一 致区 社保卡 管 理 服务中心 银行 政府 社保 卡制卡 发卡 服务 时效 严肃性 政府 方案 协议 予以 约定 银行 影 响 生育 费用 报销 拨打 社保局 咨询 热线 接通 情况 下以 力量 权利 出 发点 制卡 时效 约束 请 问 后续 局部 门否 服务 提供商 约束 只能 退而求 其次 办理 临时 社保卡	5	**办理** **社保** 证明 网上 系统 **窗口** 记录 客服 业务 查询 身份证 工作 人员 办事 **生育** 资料 居住证 缴费 保险 手续 工作 数据 养老保险 信 息 申请 排队 市人 号 码 月份 致电 **社保局** 提供 自助 告知 津贴 外省 移交 疑问 询问 市政 公众	办理 社保 窗口 生育 社保局
		11	房屋 征收 居住 补偿 **方案** 评估 上级领导 地 点 年月日 电线 工作人 员 实地 照片 号房 装 饰 彻查 装修 属实 申 请人 申请 **权利** 补偿款 作出 损失 拆迁 造假 全家 位于 导致 请求 入户 **费用** 配合 产权 人民政府 设施 价值 第 号 依法 答复	方案 权利 费用

	70	合同 条款 签订 责任 买受人 **约定** 空调 承担 损失 出卖 权益 **权利** 霸王 赔偿 交付 房屋 **协议** 侵犯 消费者 违反 侵害 违约 格式 修复 违约金 合法权益 排除 同意 逾期 有权 并未 过程 导致 受损 协商 义务 签署 请求 法律 商品房	约定 权利 协议

留言 285560 与主题 5 的重叠词汇最多，有 5 个，分别为“办理 社保 窗口 生育 社保局”,故留言 285560 所属主题为主题 5。

若某条留言与多个主题的重叠词汇都一样多，则通过 5.1.4 章节中所求出的词汇权重，将与不同主题重叠的词的词权求和，选择词权求和结果最大的那个主题为该文本所属主题。如下表所示：

表 3. 重叠词汇展示

留言编号	留言内容	主题编号	重叠主题	重叠词汇
285549	安沙镇 毛塘 工业园 一家 铝合金 生产厂 影响 环境 西地省 消防器材 有 限公司 坐落于 安沙镇 毛 塘 工业园 厂房 对外 出租 去年 增加 一家 铝合金 生 产厂家 位于 出租 仓库 位 置 靠近 周边 居民区 周边 居民 生活 带来 影响 居民	66	医院 **门口** 出口 反对 用 地 进出 **带来** 出入 设置 规划 **环境** 医疗 业主 设 立 以北 医疗机构 以南 空气 危害 后果 选址 综 合 西侧 开设 以西 山林 人员 北门 **生活** 预留 嘈 杂 桐梓 站台 以东 **污染** 不堪设想 独立 诺亚 东门 传播	门口 带来 环境 生活 污染
	代表 特留 希望 相关 部门 重视 简单 介绍 情况 厂家 生产 时间 混乱 加班 深夜 管理混乱 厂房 门口 垃圾 成堆 臭气熏天 噪音 污染 白天 晚上 切割 机械 碰撞 声声 入耳 周边 居民 带来	80	施工 扰民 **噪音** 凌晨 夜间 工地 作业 城管 噪 声 安静 连续 中海 半夜 持续 通宵 晚上 睡眠 分 贝 休息 三期 每天 **晚上** 噪声污染 **环境** 热线 睡觉 三点 **白天** 停止 十点 拨	噪音 晚上 环境 白天 周边

影响 相关 部门 给予 查处 厂家 完整 证照 手续 环评 手续 环保 设施 到位 一段 时间 厂房 飘出 很重 异味 闻到 感到 恶心 地块 地类 厂房 符合 铝合金 生产厂 家 特此 留言 给予 回复	84	打 入睡 施工单位 **周边** 开工 渣土 职能部门 运输 家里 城管局 高考 **生活 环境** 垃圾 **垃圾** 站 改善 居住 困扰 **带来** 天气 学士 **臭气熏天** 夏天 清理 水压 家住 搬迁 恳 请 频繁 舒适 文明城市 臭味 五一 身体健康 陆续 样子 朝阳 无水 美好 可 用 露天 才子 堆放 蚊虫 干净 赌博 健康 宾馆 转 入 一户 经常性	生活 环境 垃圾 带来 臭气 熏天

可以看到留言 285549 与三个主题的重叠程度都一样，则再分别求取三个主题重叠词汇的权重：

$$Weight\_1 = weight(word_1) + weight(word_2) + weight(word_3)$$

$$Weight\_2 = weight(word_1) + weight(word_2) + weight(word_3)$$

$$Weight\_3 = weight(word_1) + weight(word_2) + weight(word_3)$$

$weight(word)$表示词汇 word 在留言 285549 中的权重，由步骤三的公式计算得到。计算得出$Weight_1 < Weight_2 < Weight_3$，故留言 285549 所属的主题是主题 84。

5.1.6 相似主题合并

如前文所述，用困惑度判断主题数量是常常会导致主题数量过多、主题之间辨析度不明显的问题，为了提高文本主题归类效果，将相似主题合并[11][12][13]。

对任意主题，将该主题内所包含的所有词汇的词向量求和后再取均值，以此作为该主题的主题向量。利用余弦相似度原理，分别计算每个主题与不同主题之间的主题向量相似度。对任一主题，若其最相似主题都是彼此，则这两个主题认为是相似主题，将其合并。

表 4. 相似主题及其相似度

最相似主题	主题相似度	最相似主题	主题相似度	最相似主题	主题相似度
(0, 53)	0.849117633	(30, 38)	0.870492566	(60, 79)	0.91432851
(1, 11)	0.823316613	(31, 8)	0.864060796	(61, 45)	0.826964202
(2, 31)	0.858192425	(32, 16)	0.877261666	(62, 51)	0.842517339
(3, 59)	0.858760501	(33, 6)	0.805689667	(63, 55)	0.811734028

(4, 79)	0.891438328	(34, 67)	0.812204776	(64, 81)	0.868845841
(5, 48)	0.822757941	(35, 4)	0.812155421	(65, 72)	0.79067631
(6, 33)	0.805689667	(36, 79)	0.913940792	(66, 23)	0.816643746
(7, 8)	0.838691428	(37, 24)	0.815719205	(67, 83)	0.850300701
(8, 31)	0.864060796	(38, 85)	0.883032882	(68, 79)	0.889982146
(9, 14)	0.806380839	(39, 62)	0.835579595	(69, 41)	0.881533442
(10, 12)	0.798876692	(40, 1)	0.817477858	(70, 11)	0.819066029
(11, 58)	0.891067834	(41, 69)	0.881533442	(71, 32)	0.834134405
(12, 48)	0.832751669	(42, 73)	0.676023925	(72, 86)	0.814227315
(13, 31)	0.772076309	(43, 19)	0.840291783	(73, 29)	0.78986393
(14, 79)	0.818035031	(44, 47)	0.829540856	(74, 11)	0.83506878
(15, 71)	0.781504927	(45, 19)	0.835162338	(75, 8)	0.859357011
(16, 32)	0.877261666	(46, 50)	0.812074665	(76, 8)	0.837961701
(17, 74)	0.819532491	(47, 79)	0.891834556	(77, 56)	0.819142462
(18, 58)	0.867569666	(48, 12)	0.832751669	(78, 36)	0.74370127
(19, 58)	0.86091308	(49, 3)	0.820392966	(79, 60)	0.91432851
(20, 11)	0.823879669	(50, 46)	0.812074665	(80, 87)	0.81133894
(21, 8)	0.83990152	(51, 79)	0.868957443	(81, 64)	0.868845841
(22, 61)	0.807979844	(52, 59)	0.840579326	(82, 51)	0.792622921
(23, 81)	0.822043107	(53, 0)	0.849117633	(83, 67)	0.850300701
(24, 37)	0.815719205	(54, 14)	0.780210799	(84, 9)	0.794050213
(25, 74)	0.802680948	(55, 57)	0.864355713	(85, 38)	0.883032882
(26, 31)	0.802767759	(56, 18)	0.826649752	(86, 76)	0.832341595
(27, 89)	0.815365178	(57, 55)	0.864355713	(87, 68)	0.859940627
(28, 45)	0.827488561	(58, 11)	0.891067834	(88, 37)	0.796069221
(29, 20)	0.792299669	(59, 3)	0.858760501	(89, 69)	0.857929999

基于前文所述，相似主题有[0, 53],[3, 59],[6, 33],[8, 31],[11, 58],[12, 48],[16, 32],[24, 37],[38, 85],[41, 69],[46, 50],[57, 55],[60, 79],[64, 81],[67, 83]，分别合并这些主题以构建新的主题，主题数量由原本的 90 变为了 75，同时，也将合并的主题所对应的留言也合并在一起，如主题 3 中包含 91 条留言，主题 59 中包含 16 条留言，合并后主题所包含的留言数量为 117。

5.1.7 话题热度计算

热点的评估往往基于人们对于该热点的关注程度，而某一话题的点赞数、反对数和留言数则可直观的反映出人们对于这一话题的关注情况。根据文本所给出的标签，综合分析后提出一种融合点赞数、反对数、同类别留言数的话题热度计算方法。

$$Hot\_score = against\_num + support\_num + same\_topic\_document\_num$$

5.2 实验结果分析

经过上述步骤，将留言内容划分为 75 个类别，每个类别中分别包含数量不一的留言，具体的如下表所示：

表 5. 75 个主题及其热度指数

类别序号	留言数量	点赞数量	反对数量	热度指数	类别序号	留言数量	点赞数量	反对数量	热度指数
0	158	109	41	0.119048	38	34	26	5	0.0242
1	107	160	11	0.107728	39	57	211	1	0.107338
2	62	30	7	0.037471	40	49	52	3	0.039422
3	144	276	19	0.17018	41	26	10	0	0.012881
4	95	83	9	0.072209	42	17	3	2	0.009368
5	33	16	1	0.018345	43	66	38	3	0.040593
6	121	140	8	0.103825	44	77	48	26	0.058158
7	90	2468	7	1	45	73	30	14	0.044496
8	112	2002	10	0.82904	46	40	246	5	0.112412
9	160	144	19	0.124902	47	23	6	3	0.0121
10	69	74	11	0.062061	48	20	62	62	0.055035
11	214	2319	31	0.99961	49	14	5	2	0.007026
12	30	21	2	0.019516	50	10	8	0	0.005855
13	100	120	8	0.087822	51	16	4	0	0.007416
14	83	58	4	0.055425	52	14	33	4	0.019126
15	69	104	12	0.071038	53	45	97	1	0.054645
16	26	211	5	0.093286	54	3	0	0	0
17	4	6	0	0.002732	55	40	758	17	0.319282
18	101	107	4	0.082358	56	32	8	0	0.014442
19	87	145	12	0.094067	57	53	181	0	0.090164
20	23	9	0	0.011319	58	63	59	0	0.047229
21	18	29	1	0.017955	59	33	20	4	0.021077
22	27	55	0	0.030835	60	37	17	1	0.020297
23	7	1	0	0.007026	61	93	65	3	0.061671
24	61	181	6	0.098361	62	60	53	6	0.045277
25	51	14	3	0.025371	63	64	43	4	0.042155
26	47	41	5	0.035129	64	53	73	6	0.050742
27	15	51	2	0.025371	65	36	64	0	0.037861
28	19	4	1	0.008197	66	14	7	0	0.007026
29	134	48	11	0.074161	67	50	13	0	0.023419
30	35	35	10	0.030055	68	240	52	7	0.115535
31	37	108	9	0.058938	69	32	32	1	0.0242
32	12	6	1	0.006245	70	73	28	4	0.040203
33	35	25	1	0.022639	71	39	158	3	0.077283
34	53	68	2	0.046838	72	24	107	10	0.056987

35	34	22	1	0.021077	73	33	21	1	0.022248
36	55	286	2	0.132709	74	79	86	3	0.064403
37	66	146	17	0.088993					

上表展示了每个话题所包含的留言数量、点赞数量、反对数量以及对应的话题热度，其中，话题热度排名前五的话题分别为话题 7、话题 11、话题 8、话题 55、话题 3。

按照前文给出的话题热度计算方法，得出热度排名前 5 的热点话题，如图所示：

主题 3

主题 7

主题 8

主题 11

主题 55

图 7. 热点话题词云图

图中展示了热度排名前 5 的话题中部分高频词的词云图。可以看到，热度排名第一的主题 7 展示出的词汇主要和法律案件有关，包含了“案件”、“立案”、“受害人”等诸多与法律相关的词汇可以看出，普通民众在生活中遇到法律纠纷等问题，往往第一时间会寻求政府相关部门的帮助；热度排名第二的主题 11 展示的词汇主要是和小区业主等有关，从提出的热度词“开发商”、“业主”“安全隐患”等可以看出，该主题下的留言大多是小区业主关于房屋安全隐患的诉求；热度排名第三的主题 8 展示的词汇主要是围绕孩子的教育问题，从提出的热度词“幼儿园”、“小学”、“小区”等可以看出，该主题下的留言大多是与小区配套幼儿园、小学等与孩子上学有关的问题；热度排名第四的主题 55 展示出的词汇主要是和铁路交通有关的，从提出的热度词“高铁”、“铁路”、“小区”、“距离”等可以看出，该主题下的留言大多是关于铁路规划导致的居民受干扰问题；热度排名第五的主题 3 展示出的词汇主要是和城市内交通基础设施有关，从提出的热度

词”地铁“”出入口“”小区“”红绿灯“”路口设置“等可以看出，该主题下的留言大多是关于小区周边交通的完善建议，如地铁出入口规划和危险路口红绿灯设置等。

基于提取出的排名前 5 的话题中各自所展示出的高频词，以及五个话题中所包含的留言内容里所展示出的相关信息，构建如下热点问题表，包含热度排名、热度指数、问题描述等标签。

表 6. 热点问题表

热度排名	问题 ID	热度指数	时间范围	地点/人群	问题描述
1	7	1	2019/1/11 至 2020/1/7	西地省 居民	上当受骗等法律案件纠纷向相关部门投诉或者寻求帮助
2	11	0.99961	2019/1/1 至 2020/1/1	A 市 业主	房屋安全隐患向相关部门投诉要求开发商整改
3	8	0.82904	2019/1/14 至 2019/9/8	A 市 学生家长	向教育局寻求帮助以解决业主孩子教育问题
4	55	0.31928	2019/7/21 至 2020/1/7	A 市 小区居民	铁路规划对居民生活环境造成影响
5	3	0.17018	2019/1/12 至 2020/1/6	A 市 居民	地铁规划和危险路口红绿灯设置等交通建设问题投诉和建议

备注：热点问题留言明细表详见附件中的“热点问题留言明细表”。

六、答复质量评价

6.1 答复意见质量评价方案介绍

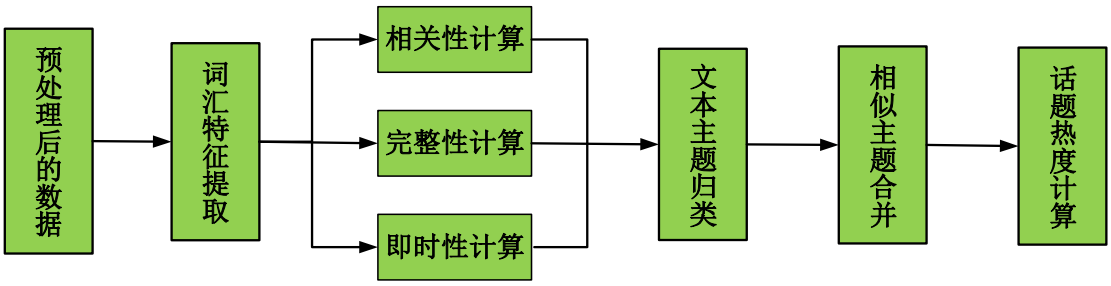

图 8. 答复意见质量评价流程图

如上图流程图所示，数据导入后，首先进行文本词汇特征提取。对于任意一条留言内容，其中所包含的各个词汇对于整条留言想反映的主题的贡献程度有区别的。这里仅提取每条留言内容和答复意见中词权靠前百分之七十的词汇用于实验，词权计算方法参照 5.1.2 中提到的多特征融合的词权计算方法。而后，量化答复质量评估因素，分别从答复意见与留言详情之间的相关性、答复意见的完整

性、答复时间的及时性三个角度进行答复质量评估，并求取任一答复意见质量的分值。

6.1.1 词汇特征提取

参照 2-3 中提到的文本词汇特征提取的方法，计算附件 4 中的留言详情的词汇权重与答复意见的词汇权重，分别提取权重靠前的百分之七十的词汇用以替代原文词汇。部分结果如下表所示：

表 7. 词权靠前词汇提取(留言详情)

留言编号	留言详情	词权靠前百分之七十
4544	市公品 商城 定点 采购 资产 评估 服务供应商 依法 公开 招投标 程序 政府 采购 电话 咨询 财政局 相关 人员 解释 公品 商城 资产 评估 服务供应商 采用 财政局 政府 投资 项目 国有资产 处置目的 招标 建立 中介机构 定点 库公品 商城 定点 采购 覆盖 政府 付费 中介 服务 建立 定点 服务内容 相差 甚远 废除 公平 商定 采购 市场 公平竞争 环境 市公品 商城 定点 采购 中介 服务供应商 覆盖 政府 付费 中介 服务 中介 服务 政府 采购 工程造价 服务 涵盖	定点 采购 商城 服务 资产 评估 公平 服务供应商 中介 市公品 财政局 付费 覆盖 公品 中介机构 库公品 商定 政府 工程造价 国有资产 建立 招标 程序 涵盖 目的 内容 依法 市场 废除 电话
11538	社区 红旗区 居民 天然气 实事 一件 大好事 天然气 户外 工程 管线 房子 端头 几个 不知 何故 动静 老百姓 期盼 好事 半拉子 工程 希望 社区 天然气 送入 居民 家中	天然气 红旗区 进户 多久 端头 半拉子 管线 社区 工程 大好事 动静 送入 实事 好事 居民 户外 何故

表 8. 词权靠前词汇提取(答复意见)

答复意见编号	答复意见	词权靠前百分之七十
6488	网友 您好 留言 收悉 情况 回复 涉及 租赁 合同 条件 落户 务工人员 落户 一项 常住 户口 登记 管理 务工人员 落户 登记 条件 第二项 县区 市县 合法 稳定 住所 租赁 人员 申请 居住 生活 配偶 子女 父母 户口 迁入 居住地 城镇 地区 派出所 办理 前往 租赁 房屋 所在 辖区 公安 派出所 办理 疑问请 来电 咨询 公安局 人口 出入境 管理 支队 感谢您 工作 支持 理解 监督 年月日	落户 务工人员 咨询 租赁 户口 派出所 住所 县区 条件 居住地 市县 支队 城镇 父母 子女 地区 公安局 辖区 公安 人口 合法 办理 登记 房屋 配偶 来电 常住 人员 迁入 疑问请 涉及 管理 居住 情况 网友 稳定 第二项

20276	网友 您好 留言 收悉 回复 漓楚路 以南 三景 国际 小区 武警 基地 之间 闲置地 规划 商业 用地 巨星 地产 公司 摘牌 桩基础 武警 基地 尚未 搬迁 暂未 动工 长丰 集团 总部 和星 国际 小区 之间 商住 用地 面积 咨询 建设局 博雅 路灯 暂未 纳入 建设 计划 街道 协调 博雅 早日 路灯 建设 感谢您 信任 欢迎您 一如既往 理解 支持 监督 工作 泉塘 街道 办事处	泉塘 规划 咨询 博雅 基地 路灯 闲置地 巨星 国际 武警 三景 桩基础 和星 漓楚路 用地 地产 长丰 总部 信任 小区 动工 街道 摘牌 建设局 商业 集团 办事处 面积 计划 商住 公司 之间 暂未 搬迁 建设 网友

可以看到，基于词权提取关键词后的留言内容和答复意见都更加凝练，减少了一些词权低、即干扰词汇的影响。如留言编号为 11538 原文中出现的“几个”、“不知”、答复意见编号为 6488 中的“网友”、“您好”，这类词汇在原文中的重要性较低，去除后并不影响整体主题效果。

6.1.2 答复质量评估因素量化

（1）相关性量化

对于任意一条留言，对其所包含的每个词取词向量后，将每个词的词向量求和后再取平均值，以此作为这条留言内容的留言向量；再以同样的方法来计算每条答复意见的答复向量。计算每条留言向量与答复向量之间的相似度，以此作为留言详情与答复意见的相关性评估指标。

$$vector(\text{留言详情}) = \frac{vector(word_1) + vector(word_2) + \cdots + vector(word_n)}{n}$$

$$vector(\text{答复意见}) = \frac{vector(word_1) + vector(word_2) + \cdots + vector(word_m)}{m}$$

$$Relativity = \text{cosine\_similarity}\langle vector(\text{留言详情}), vector(\text{答复意见})\rangle$$

其中$word_n$、$word_m$分别指某条留言详情与答复意见中的词汇，n和m分别表示该条留言详情与答复意见中词汇的数量，但是每一条留言与答复中的词汇数量都可能是不一样的，故n和m并不固定。cosine_similarity指将留言详情向量与答复意见向量计算余弦相似度。

（2）完整性量化

按 2.1 的方法求得每条留言的留言向量；计算答复意见中每个词的词向量与对应留言的留言向量之间的相似度，若相似度超过特定阈值，则认为答复意见中的这个词汇与留言详情具有较大关联性。而答复意见中这种与留言详情具有较大关联性的词汇数量占总词汇数量的比重越大，则说明这个答复越完整。具体算法如下所示：

Algorithm 1 Integrity Algorithm

Input: FModel,Response,Message

Output: Integrality

function "IntegralityCalculated"
  $count = 0$
  $T = 0$
  $MessageVector = mean(sum(FModel(word_{i,j})))i = 1, 2, ..., n; j = 1, 2, ..., m$
  $ResponseWordVector = FModel(word_{i,k})k = 1, 2, ..., d$
  while $T < m$ **do**
    $Sim = CosineSimilarity[MessageVector, ResponseWordVector]$
    if $Sim > ThresholdValue$ **then**
      $count+ = 1$
    end if
    $T+ = 1$
  end while
  $Integrity = count/length(ResponseWord)$
end function

图 9. 完整性计算过程伪代码

其中，FModel 表示用维基百科语料库和此次实验数据集训练出的 FastText 词向量模型；Response 表示预处理后的答复意见；Message 表示预处理后的留言内容；Integrity即表示答复完整性指数；Messagevector 表示留言向量，是通过留言中的词向量求和后再取均值得到的；Respondwordvector 表示答复意见中某个词汇的词向量；wordij 表示第 i 条留言中的第 j 个词汇，该条留言的词汇数量为 m；wordik 表示第 i 条答复意见中的第 k 个词汇，该答复意见中词汇数量为 d，值得一提的是，不同的留言和答复意见中的词汇数量 m、d 是不一样的；

Sim 表示答复意见中的任一词汇与留言向量之间的余弦相似度；ThresholdValue 表示设定的比较阈值；若答复意见中与留言详情的相似度超过阈值的词汇数量越多，表示该条答复越完整；此次设定相似度阈值为 0.35。

（3）及时性量化

附件 4 中包含了留言时间标签、答复时间标签，实验中求取二者的时间间隔、并新添加了一个“时间间隔”标签，如下图所示：

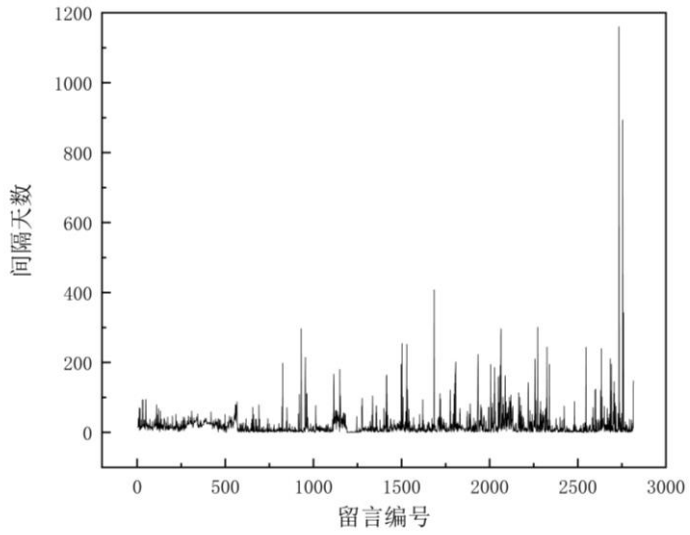

图 10. 留言答复时间间隔标签图

此次实验认为答复时间与留言时间的时间间隔越小，说明答复意见的即时性越好，如上图，有些答复意见的答复时间与留言时间仅间隔几天，而有些却间隔数月甚至数年。

6.1.3 答复质量分值计算

融合上述三个因子提出一个答复质量分数计算方法。

$$\text{RespondQualityScore} = \alpha * \text{Relativity} + \beta * \text{Integrity} - \text{ImmediateNormalization}$$

表示答复质量分数，其中，Relativity表示相关性指数，Integrity表示完整性指数，这里设置各自的参数为α为 0.6，β为 0.4； ImmediacyNormalization表示经过正则化后的时间间隔，时间间隔越小及时性越好，故这里减去时间间隔正则化后的结果。

6.2 答复意见质量评估实验结果分析

6.2.1 实验流程分析

为了有效评估每条答复意见的质量，结合了答复意见与留言内容之间的相关性、答复意见对于留言内容的完整性以及答复意见的及时性三个指标来综合分析每条答复意见，将各个评估指标量化并生成答复质量评分表，部分结果如下表所示：

表 8. 答复质量评价示例

序号及指标	留言详情	答复意见
15	尊敬的领导:泉塘街道漓楚路和小塘路交汇处车流量、人流量密集,无过街天桥和地下通道，交通事故较多，此处无高杆灯一到晚上视线较差,小塘路以南至盼盼路沿线路灯较差，晚上老人、小孩过街极为不方便。希望借城市提质改造机会能在漓楚路和小塘路交汇增加高杆灯,改造小塘路路灯照明，提高城市品质，增强老百姓幸福感和获得感。	您好！您在信中提到的关于“泉塘街道漓楚路和小塘路交汇处车流量、人流量密集，无过街天桥和地下通道，交通事故较多，此处无高杆灯一到晚上视线较差，小塘路以南至盼盼路沿线路灯较差，晚上老人、小孩过街极为不方便。希望借城市提质改造机会能在漓楚路和小塘路交汇增加高杆灯，改造小塘路路灯照明，提高城市品质，增强老百姓幸福感和获得感。”的问题，我局高度重视，现回复如下：2017 年我局已考虑在漓楚路和小塘路安装高杆灯，但由于该路口处于高压配送网正下方，该路口无法安装高杆灯。小塘路设计之初根据道路宽度是安装的单侧路灯，根据实际需要，我局会考虑将该路段路灯增补纳入到下一年度改改造计划中。感谢您对城市管理工作的关注，今后如您再次遇到城市管理相关问题，欢迎致电城管服务热线 0000-00000000 或 12319，我局将热忱为您服务。2019 年 5 月 14 日
相关性	0.941277231	
完整性	0.695652174	
即时性	0.004844332	
质量分值	0.838182876	

表 9. 答复质量评价示例

序号及指标	留言详情	答复意见
2808	我是 2018 级新生家长，H 市学院在军训期间无理收费，两双军训袜子，1 双鞋垫，1 个本子和一支笔共计 45 元，45 元是我孩子 3 天的生活费，学院如此这般威逼，希望有关部门能处理。	您好，您所反映的问题，已转交相关部门调查处置。
相关性	0.308715273	
完整性	0	
即时性	0.003249177	
质量分值	0.181979987	

如上表，序号 15 中的答复中，详细的为留言者分析问题、解答问题，故相关性和完整性都比较好，而序号 2808 中的答复中，并没有对留言者的问题作直接答复，而是将相应问题转交到其他职能部门处理，故其相关性和完整性都比较差。

七、总结

本文基于多种自然语言处理方法，针对附件中给出的群众留言内容和相关部门的答复意见数据，完成了留言分类、留言热点问题挖掘以及答复质量评价指标的构建三个任务。对于留言分类任务，与传统方法中利用 TFIDF 算法来构建文本向量空间模型不同的是，文中提出了一种融合多种词汇特征的文本向量空间构建方法，在减少词袋长度的同时也取得了较好的文本分类效果。对于热点问题挖掘任务，文中通过 LDA 主题模型实现对留言内容的主题提取和文本聚类，再利用 FastText 词向量模型对相似主题合并，并且提出一种话题热度计算方法、提取出热度排名前五的话题。对于相关部门答复意见的质量评估，文中从答复与留言之间的相关性、答复对于留言的完整性以及答复的及时性三个角度构建了一个答复意见质量评估方法。

最后十分感谢泰迪杯官方给出关于“智慧政务”中文本挖掘应用的赛题，通过本次数据挖掘项目分析，我们最大的收获是对利用支持向量机、LDA主题模型和FastText词向量模型来分析本赛题，从而对数据分类的过程有了更深的认识与感触。我们深深意识到一个好的模型的建立并非一蹴而就，而是在不断的试探过程中得到的。

参考文献

[1] 薛苏琴,牛永洁.基于向量空间模型的中文文本相似度的研究[J].电子设计工程,2016,24(10):28-31.

[2] 施聪莺,徐朝军,杨晓江.TFIDF 算法研究综述[J].计算机应用,2009,29(S1):167-170+180.

[3] Liu X, Zhang Z, Li B, et al. Keywords Extraction Method for Technological Demands of Small and Medium-Sized Enterprises Based on LDA[C]//2019 Chinese Automation Congress (CAC). IEEE, 2019: 2855-2860.

[4] 高楠,李利娟,李伟,祝建明.融合语义特征的关键词提取方法[J].计算机科学,2020,47(03):110-115.

[5] Cortes C, Vapnik V. Support vector machine[J]. Machine learning, 1995, 20(3): 273-297.

[6] Bojanowski P, Grave E, Joulin A, et al. Enriching word vectors with subword information[J]. Transactions of the Association for Computational Linguistics, 2017, 5: 135-146.

[7] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[8] 关鹏,王曰芬.科技情报分析中 LDA 主题模型最优主题数确定方法研究[J].现代图书情报技术,2016(09):42-50.

[9] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(Jan): 993-1022.

[10] 孙明珠,马静,钱玲飞.基于文档主题结构和词图迭代的关键词抽取方法研究[J].数据分析与知识发现,2019,3(08):68-76.

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

[12] 汪静,罗浪,王德强.基于 Word2Vec 的中文短文本分类问题研究[J].计算机系统应用,2018,27(05):209-215.

[13] 牛雪莹,赵恩莹.基于 Word2Vec 的微博文本分类研究[J].计算机系统应用,2019,28(08):256-261.

附录

附件 2 留言内容分类准确率

实验编号	随机种子	交通运输	劳动和社会保障	卫生计生	商贸旅游	城乡建设	教育文体	环境保护
1	53	0.917241	0.932039	0.95	0.900332	0.888676	0.954545	0.938865
2	99	0.925373	0.936508	0.945813	0.893688	0.864407	0.947631	0.944206
3	38	0.93007	0.906977	0.941748	0.875421	0.882576	0.959184	0.946667
4	32	0.892086	0.930097	0.957143	0.889273	0.872694	0.96144	0.964126
5	39	0.933333	0.93988	0.951691	0.89899	0.884112	0.950372	0.95671
6	77	0.932836	0.941532	0.934884	0.921233	0.871795	0.962217	0.938326
7	86	0.921429	0.945122	0.926941	0.889262	0.878731	0.954774	0.96875
8	52	0.916667	0.956701	0.896396	0.87	0.877095	0.956633	0.951542
9	55	0.902778	0.932271	0.951923	0.88746	0.896686	0.962312	0.943723
10	53	0.917241	0.932039	0.95	0.900332	0.888676	0.954545	0.938865
均值		0.918905	0.935317	0.940654	0.892599	0.880545	0.956365	0.949178

附件 2 留言内容分类召回率

实验编号	随机种子	交通运输	劳动和社会保障	卫生计生	商贸旅游	城乡建设	教育文体	环境保护
1	53	0.863636	0.973631	0.863636	0.891447	0.920477	0.949749	0.914894
2	99	0.805195	0.957404	0.872727	0.884868	0.912525	0.954774	0.93617
3	38	0.863636	0.94929	0.881818	0.855263	0.926441	0.944724	0.906383
4	32	0.805195	0.971602	0.913636	0.845395	0.940358	0.939698	0.914894
5	39	0.818182	0.951318	0.895455	0.878289	0.940358	0.962312	0.940426
6	77	0.811688	0.947262	0.913636	0.884868	0.946322	0.959799	0.906383
7	86	0.837662	0.943205	0.922727	0.871711	0.936382	0.954774	0.923404
8	52	0.857143	0.941176	0.904545	0.858553	0.936382	0.942211	0.919149
9	55	0.844156	0.94929	0.9	0.907895	0.914513	0.962312	0.92766
10	53	0.863636	0.973631	0.863636	0.891447	0.920477	0.949749	0.914894
均值		0.837013	0.955781	0.893182	0.876974	0.929423	0.95201	0.920426

附件 2 留言内容分类 F 值

实验编号	随机种子	交通运输	劳动和社会保障	卫生计生	商贸旅游	城乡建设	教育文体	环境保护
1	53	0.889632	0.952381	0.904762	0.895868	0.904297	0.952141	0.926724
2	99	0.861111	0.946841	0.907801	0.889256	0.887814	0.951189	0.940171
3	38	0.895623	0.927651	0.910798	0.865225	0.903977	0.951899	0.926087
4	32	0.846416	0.950397	0.934884	0.866779	0.905263	0.950445	0.938865
5	39	0.871972	0.945565	0.922717	0.888519	0.911368	0.956305	0.948498
6	77	0.868056	0.944388	0.924138	0.902685	0.907531	0.961006	0.922078
7	86	0.877551	0.944162	0.924829	0.880399	0.906641	0.954774	0.945534
8	52	0.885906	0.948875	0.900452	0.864238	0.905769	0.949367	0.935065
9	55	0.872483	0.940704	0.925234	0.897561	0.905512	0.962312	0.935622
10	53	0.889632	0.952381	0.904762	0.895868	0.904297	0.952141	0.926724
均值		0.875838	0.945334	0.916038	0.88464	0.904247	0.954158	0.934537