“智慧政务”中的文本挖掘应用

摘要

近年来，网络问政平台的地位日益凸显，公众留言数量激增，给传统人工处理带来巨大压力。因此，建立智慧政务系统是社会治理创新发展的新趋势。本文旨在利用自然语言处理和文本挖掘技术对网络问政平台进行研究，对群众的海量留言进行标签分类，挖掘出留言反映的热点问题以及针对政府答复意见进行全方位评价。

针对问题一，本文先对留言详情进行数据预处理操作，后将留言文本与 7 种标签进行数字化表示，对留言文本进行修剪及填充。基于上述操作，本文选择深度学习下的 CNN 卷积神经网络模型。结合 TensorFlow 框架，通过构建恰当的网络结构和调整超参数训练模型，进行留言的标签分类。最终模型的 F1-score 分数稳定于 0.88，留言分类效果较好。

针对问题二，文本结合先前留言详情的数据预处理操作，对其进行“地点”词频增强、TF-IDF 特征提取及 PCA 降维。通过 DBSCAN 聚类算法初步筛选出多个热点话题后，文本利用构建的四个评价指标进行灰色关联分析输出指标相应权重，最终通过计算热度指数筛选出排名前五的热点话题。

针对问题三，本文在对留言详情及答复意见进行数据预处理的基础上，从相关性、完整性和可解释性对答复意见进行评价。对于相关性，本文选择使用余弦相似度进行衡量；对于完整性，基于 Stanford NLP 可实现的句法分析，本文构建三个具体公式进行完整性的计算；对于可解释性，本文通过判断相关法律法规文件是否出现作为衡量标准。最终得到三个性质的分值对政府答复意见进行分析与评价。

关键词：CNN；TensorFlow；TF-IDF；PCA 降维；DBSCAN 聚类分析；灰色关联分析；余弦相似度；Stanford NLP

Application of text mining in "intelligent government"

Summary

The purpose of this paper is to use natural language processing and text mining technology to study the network political platform, to label and classify the massive messages of the masses, to mine the hot issues reflected in the messages, and to make an all-round evaluation of the government's reply opinions.

Aiming at the first problem, this paper first carries out data preprocessing operation on the message details, then digitally represents the message text with 7 kinds of labels, and trims and fills the message text. Based on the above operations, this paper selects CNN convolution neural network model under deep learning. Combined with TensorFlow framework, the label classification of messages is carried out by constructing an appropriate network structure and adjusting the super-parameter training model. The F1-score score of the final model is stable at 0.88, and the message classification effect is good.

To solve the second problem, the text combines the Data pre-processing operation of previous message details to enhance the "location" word frequency, extract the TF-IDF feature and reduce the dimension of PCA. After several hot topics were selected by DBSCAN clustering algorithm, the text uses the four evaluation indexes to carry out the corresponding weight of the output index of grey relational analysis, finally through the calculation of heat index sifted out the top five hot topics.

In response to the third question, on the basis of data preprocessing of message details and reply opinions, this paper evaluates the reply opinions from relevance, integrity and interpretability. For correlation, this paper chooses cosine similarity to measure it. For integrity, based on the syntactic analysis that Stanford NLP can implement, this paper constructs three specific formulas to calculate integrity. For interpretability, this paper judges whether relevant laws and regulations appear as a measure. Finally, the scores of three natures are obtained to analyze and evaluate the government's reply opinions.

Keywords: CNN; Tensorflow; TF-IDF; PCA dimensionality reduction; DBSCAN Cluster Analysis; Grey Relational Analysis; cosine similarity; Stanford NLP

目录

1 挖掘目标

本次建模目标是利用收集自互联网公开来源的群众问政留言记录，及相关部门对部分群众留言的答复意见，利用人工智能自然语言处理，综合评价以及变量定义等方法，达到以下三个目标：

（1）按照一定的划分体系建立关于留言内容的一级标签分类模型对留言进行分类，以便后续将群众留言分派至相应的职能部门 处理。

（2）利用群众留言中的相关指标，通过综合评价选出“热门问题”，并提取出问题发生的地点信息，有助于相关 部门进行有针对性地处理，提升服务效率。

（3）通过对相关性、完整性、可解释性的合理定义，从答复的这三个指标等角度对答复意见的质量给出一套评价方案。

2 问题一的分析过程与结果展示

2.1. 数据预处理

根据数据科学中“Garbage In， Garbage Out”的原理，对不满足数据质量要求的数据进行分析，往往得到的是不准确甚至错误的结论。这些“脏数据”将不同程度地影响数据挖掘的效率，因此数据清洗是整个数据挖掘过程中非常重要、不可缺少的一个环节，其结果质量直接关系到后续的数据挖掘分析以及最终结论。

因此，在对群众留言进行分类前，我们需要先对留言详情进行 jieba 分词、去除停用词、歧义词的处理。

(一) jieba 中文分词

中文分词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。它以词作为基本单元，使用计算机自动对中文文本进行词语的切分，即让词之间以空格分隔，方便计算机识别出各语句的重点内容。因此，通过分词操作，我们将文档中的文本转换成词条，使得这些词条成为未来特征词条的主要来源。

本文主要研究对象是用户留言，在中文文词技术上我们将进行如下处理：

1、 词语切分

我们采用 python 的中文分词包 jieba 进行分词。jieba 采用了基于前缀词典

实现的高效词图扫描，生成句子中汉字所有可能成词情况所构成的有向无环图(DAG)，同时采用了动态规划查找最大概率路径,找出基于词频的最大切分组合。以其中一条留言为例，分词效果举例如下所示：

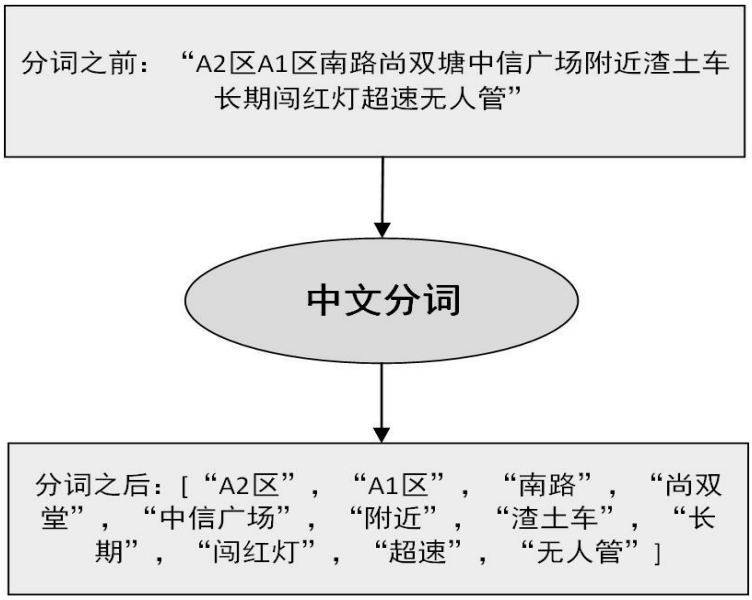

图 1： 中文分词演示

2、 未登录词识别

未登录词即没有被收录在分词词表中但必须切分出来的词，包括各类专有名词（人名、地名、企业名等）、缩写词、新增词汇等等），对于未登录词，jieba 分词系统采用了基于汉字成词能力的 HMM 模型，会通过 HMM 的方式对未登录词进行标注。使得中文分词效果更好地呈现。

3、 词性标注

词性（词类）是词汇的基本语法属性，而词性标注是在给定句子中判定每个词的语法范畴，确定它的词性并加以标注的过程。jieba 分词系统的词性标注功能是在 jieba/posseg 目录下实现的。导入 jieba.posseg 后，我们实现了对用户留言的词性标注过程。

词性标注举例如下所示：

“恳请政府帮助解决 A8 县未来方舟办理房产证的难题”的词性标注结果为：“恳请/v 政府/n 帮助/v 解决/v A8 县/n 未来/t 方舟/n 办理/v 房产证/j 的/uj 难题/n”。

（二） 停用词及消歧词词典处理

停用词指没有实际含义、不作为结果的词，如冠词、介词、副词或连词等。此外，中文分词器切分出来的所有词条中含有大量的单个独立字。经过研究发现，这些单个独立字不仅所携带的文本信息量较少，而且还对其他实词起到一定的抑制作用，降低了分类系统的处理效率和准确度，因此，文本预处理过程有必要将所有的单个独立字过滤。

通常，中文停用词分为如下几类：

普通标点符号及特殊符号。

留言中的普通标点符号及特殊符号对于文本内容起到分割句子、表达用户情绪、传达用户个人情感等作用，它们并未携带有价值的文本信息，因此对于文本处理而言，它们应当被剔除。

无意义的虚词。

如“你们”、“我们”、“他们”、“可以”等，虚词在文本中的出现频率很高，对于文本分析不仅会占用存储空间，降低搜索效率，还会误导我们词频越高重要性越高，事实上它们本身并无实际含义。

限定词

如“的”、“一个”、“这”、“那”等。这些词语的使用较大的作用仅仅是协助一些文本的名词描述和概念表达，并没有太多的实际含义。

除此之外，停用词还有很多其他属性。根据中文停用词词典，我们将分词结果中的大量停用词逐一筛选并剔除。

对停用词进行过滤之后，我们还需要对消歧词进行筛选和处理。由于不同的语言单位，语义分析的任务各不相同。在词的层次上，语义分析的基本任务是进行词义消歧。词义、句义以及篇章含义层次都会根据不同的上下文环境产生不同的意义，消歧就是指根据上下文确定对象语义的过程。我们计算语义词典中各个词义的定义与上下文之间的覆盖度，选择覆盖度最大的作为该词在其上下文下的正确词义。

进行 jieba 中文分词和停用词、歧义词处理后，我们初步完成留言详情的数据预处理阶段。在进行 CNN 分类模型的建立前，我们需要对留言作出如下处理：

（三） 文本的数字化表示

计算机是无法直接理解中文并对其进行运算的，我们需要将留言中的文字用数字表示，将每个文本转化为一个整数序列，忽略了词与词之间的关系、语法、语序，仅仅把文本当作是一个词集合。模型会为每个词指定一个唯一的编号 ID，如果文本中出现了这个词，那么词集合里就会有这个编号 ID，也就是

说：一条文本最后会被表征成一个数字列表，每个数字都对应着相应的词汇。举个例子，假设有两条留言：

表 1： 留言详情分词

留言详情	分词后的留言
我爱 NLP	{“我”,“爱”,“NLP”}
NLP 非常有意思	{“NLP”,“非常”,“有”,“意思”}

为了给每个词一个唯一的编号，我们需要建立一个词汇表，表中包含了留言中出现过的所有词汇：

词典={“我”,“爱”,“NLP” ,“非常”,“有”,“意思”}

并为每个词分配一个唯一的 ID：

词典={1：“我”，2：“爱”，3：“NLP” ，4：“非常”，5：“有”，6：“意思”}

这样我们的两条留言就可以被表示成：

表 2： 留言详情词典表征

留言详情	处理后的留言
我爱 NLP	[1,2,3]
NLP 非常有意思	[3,4,5,6]

数字 1 代表词汇“我”，数字 2 代表词汇“爱”，数字 3 表示词汇“NLP”，所以留言“我爱 NLP”就被表示成数字列表[1,2,3]，其他以此类推。

（四） 填充（padding）与修剪（truncating）

所有的留言用数字列表表示后会存在长短不一的情况，有的留言包含的词汇多，那么这个列表就会比较长，而包含词汇少的留言表示后的列表就会比较短。而我们后续的计算需要每条留言列表长度一致，所以就要将短列表进行填充，将长列表进行修剪。

指定一个 max_len，表示固定的列表长度，所有长度大于 max_len 的列表都要被修剪，可以从前修剪，也可以从后修剪，具体情况要根据效果而定，比如列表[3,4,5,6]，它的长度是 4，我们指定 max_len=3，如果从前修剪，那么就会得到列表[4,5,6]，3 被剪掉了；如果从后修剪，就会得到列表[3,4,5]，6 被剪掉了。同理所有长度小于 max_len 的列表都会被填充，填充就是补零，直到列

表长度等于 max_len，可以从前补零，也可以从后补零，这里不再赘述。

（五） 类别的数字化表示

数据中共有城乡建设、卫生计生等 7 种标签，与文本一样，我们也是需要将每种标签对应唯一的 ID，因此我们得到了如下字典：

类别字典={1: '城乡建设', 2: '劳动和社会保障', 3: '教育文体', 4: '商贸旅游', 5: '环境保护',6: '卫生计生',7: '交通运输'}

2.2. CNN 分类模型建立

文本分类一直是自然语言处理领域最活跃的研究方向之一，分类方法众多，经过不断比较，最终我们采用了 CNN 深度学习模型。

为了提高分类准确性，输入的数据各维度的范围应该一致，这样神经网络的才能更好的完成工作，而我们现在得到的文本向量仅仅是包含了词 ID 的列表，词 ID 的范围是从 1 到 60000 的，显然不符合我们的标准，因此我们需要在模型中加上一层嵌入层。

嵌入层其实是简单的神经网络，分为输入层、隐藏层、输出层。从输入层输入我们准备好的留言词 ID 列表，经过不断地训练，就会在输出层得到每个词的词向量，词向量的维度V可以由我们指定。

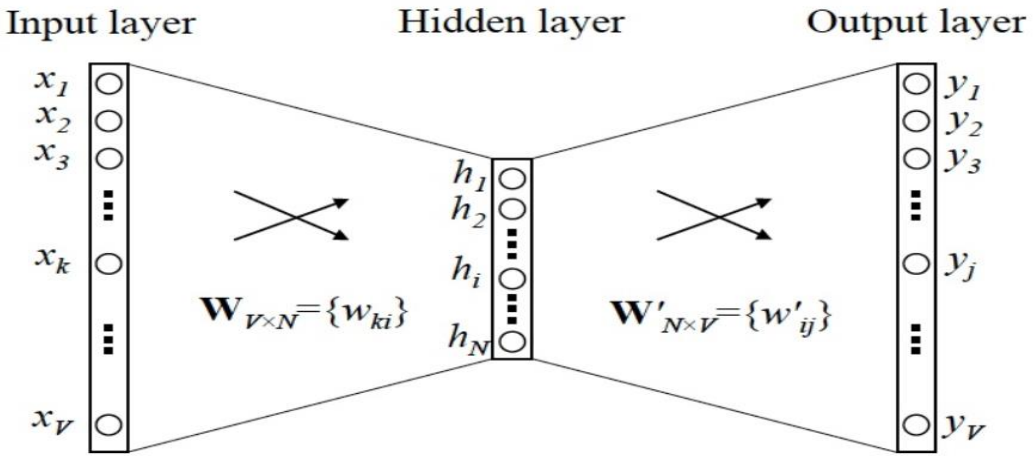

图 3： 嵌入层结构图

例如苹果=[0.98,0.94,-0.02]，香蕉=[0.96,0.04,0.97]，这里我们指定的维度是 3，这样每个词都会对应一个 3 维的词向量，每个位置的数字都代表着一定的含

义。就这个例子而言，通过对比我们可以猜测第一维数字代表水果，因为苹果和香蕉都是水果，而它们的第一维数字都非常接近 1；第二位数字可能代表红色，因为苹果接近 1，而香蕉几乎为 0；同理第三维可能是黄色。

这样就可以通过计算两个词向量的距离来判断这两个词语义是否相近。我们对题目提供的数据进行训练，得到了约 63000 个词向量，就“打扰”这个词来说，与它较相似的词如下：

表 3： 与“打扰”语义相近的词

词汇	距离
扰民	0.401
打搅	0.564
干扰	0.567
影响	0.573
说服	0.683
伤害	0.706

从表中的距离数值列可以看到“扰民”是与“打扰”最为相似的词，其次是“打搅”、“干扰”、“影响”。

词向量的可视化效果如下所示：

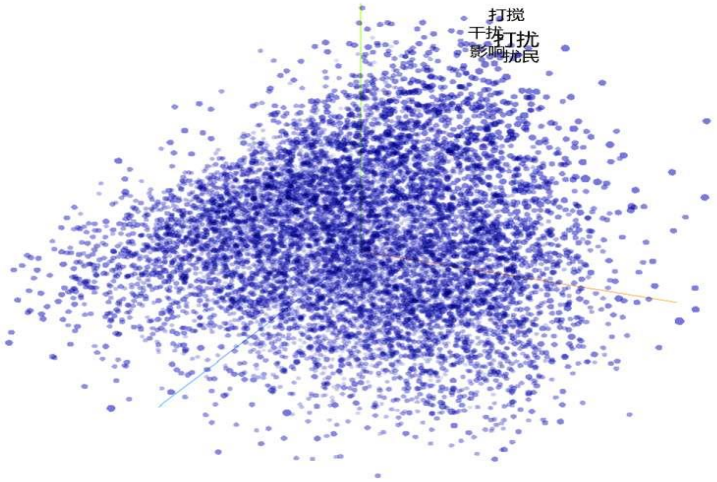

图 4： 词向量可视化图

卷积层是模型的核心，前面的嵌入层会将留言以矩阵形式（矩阵的每一行代表一个词向量）输入到卷积层，卷积层再通过卷积核的一维滑动提取文本特征。

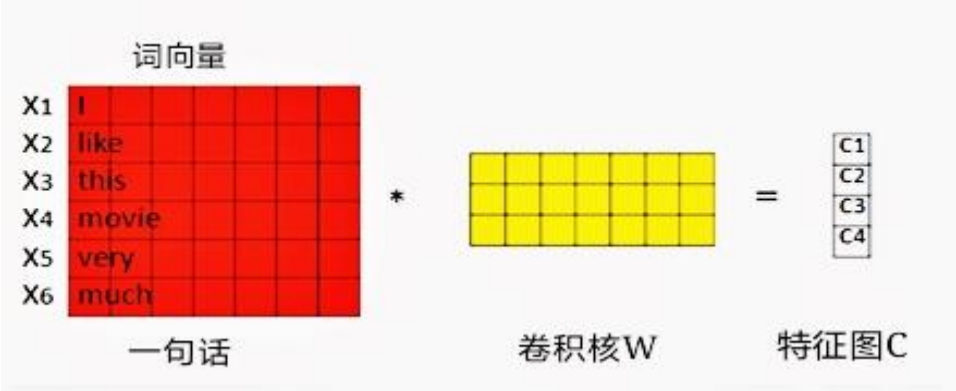

图 5： 卷积层结构图

卷积核的列数 c 等于词向量的维度，行数 r 通常取 2-8，表示每次扫过单词的个数，这样得到的特征a_i就等于：

$$C_i = \sum_{m=0}^{r} \sum_{n=0}^{c} w_{in} \times x_{mn} \quad (1)$$

一个卷积层内会有多个卷积核，每个卷积核在训练过程中它们的权重会有所差异，理想化状态下每个卷积核都会学习到文本的不同特征，可能这个卷积核学习到了文本发生的地点，另外一个卷积核学习到了文本描述的事件。卷积核的数量往往需要我们不断地调整，以达到最好的分类效果。

为了防止训练的过拟合，需要在卷积层后加上一层 Dropout，Dropout 层在每次训练时会随机舍弃掉一部分卷积核，让这部分卷积核不参与训练，相当于变成了原来网络的子网络，不仅可以大大减少计算量，还可以一定程度上防止过拟合问题的发生。

常用的池化层分为最大池化层和平均池化层两种，我们将使用最大池化层。池化的特点就是在保留显著特征的同时，降低数据的维度，输出一个固定大小的矩阵，在自然语言处理中，池化层的工作方式如下：

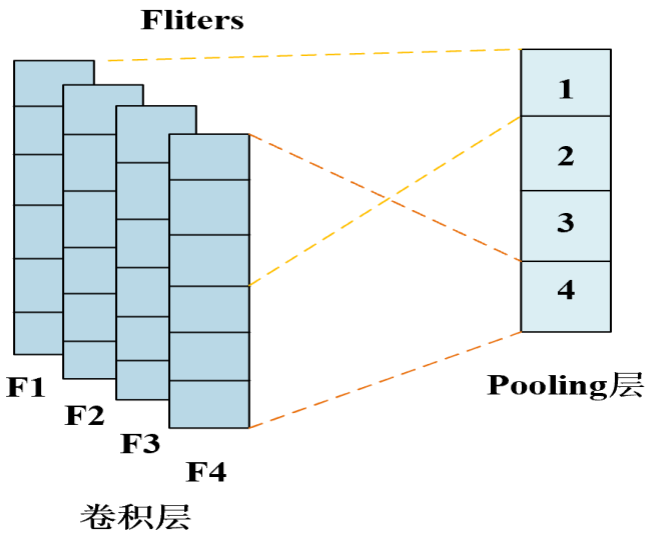

图 6： 池化层工作方式图

可以看到，最大池化层会保留每个卷积核中最显著（数值最大）的特征，在 CNN 中采用池化层可以很好的降低计算的复杂度，但与之带来的缺点就是特征其出现的位置信息并没有保留，可能会为我们的分类带来不好的影响。

到此我们已经提取了留言中的特征，所有特征最终再通过全连接层和包含 7 个神经元（留言共有 7 个类别）的 softmax 层后，输出留言类别。模型结构大致如下图所示：

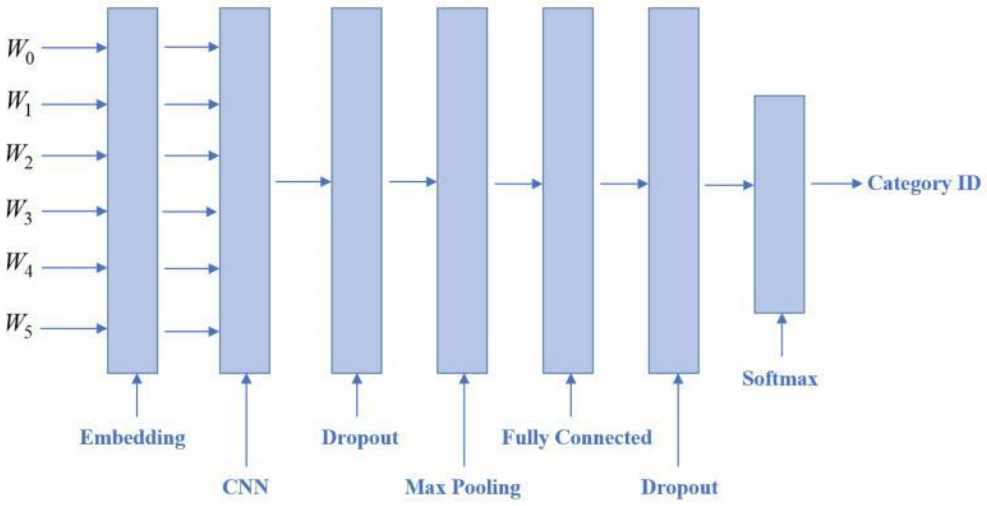

图 7： CNN 模型结构图

可以看到模型共有 7 层，分别是嵌入层、卷积层、Dropout 层、池化层、全连接层、Dropout 层、输出层。

模型涉及的参数如下表所示：

表 5： 模型参数表

Layer (type)	Output Shape	Param #
embedding_23 (Embedding)	(None, 185, 64)	1280000
conv1d_23 (Conv1D)	(None, 179, 64)	28736
dropout_46 (Dropout)	(None, 179, 64)	0
global_max_pooling1d_23(Glo)	(None, 64)	0
dense_46 (Dense)	(None, 16)	1040
dropout_47 (Dropout)	(None, 16)	0
dense_47 (Dense)	(None, 7)	119

2.3. 模型训练与结果分析

在训练前我们需要一个好的工具来帮助我们快捷地实现网络模型，而谷歌的 TensorFlow 框架在近年来逐渐成为潮流，特别是随着去年 TensorFlow2.0 的发布，让深度学习门槛变得越来越低，越来越易用，我们接下来的训练都会在 Python3.7，Tensorflow2.0.0 rc0 的环境下完成。

深度学习的效果好坏很大程度上取决于网络的结构和超参数的设置，上文已经提出了网络结构，接下来我们就需要不断调节超参数，使分类获得较好的效果，大体的步骤如下：

(一)将留言按 7：3 的比例分为训练集和验证集。

(二)指定词汇表大小 vocab_size、词 ID 列表长度 max_length、词向量维度 embedding_dim，卷积层中的卷积核数量 kernel_size、卷积核大小 strides、Dropout 层的删除比例 rate、全连接层的神经元个数 units、学习率大小 a、激活函数的选择 activation。

(三)设置训练次数 epoch，训练模型。

(四)比较训练集和验证集上的损失值 loss 和准确率 accuracy。如果分类效果不佳，则回到步骤 2 重新调整参数；如果分类效果较好，则到步骤 5。

(五)保存模型，用于以后的使用。

经过不断地尝试，最终我们设置词汇表大小 vocab_size=20000，词 ID 列表长度 max_length=185，词向量维度 embedding_dim=64，卷积层中的卷积核数量

kernel_size=64，卷积核大小 strides=7，Dropout 层的删除比例 rate=0.5，全连接层的神经元个数 units=16，学习率大小 a=0.001，卷积层的激活函数选择 relu 函数，全连接层的激活函数选择 tanh 函数。

经过 10 轮(epoch)的训练，最终结果如下图所示：

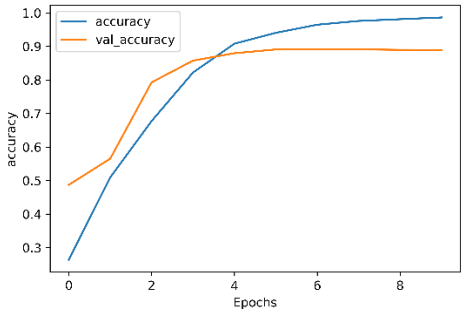

图 8： 准确率

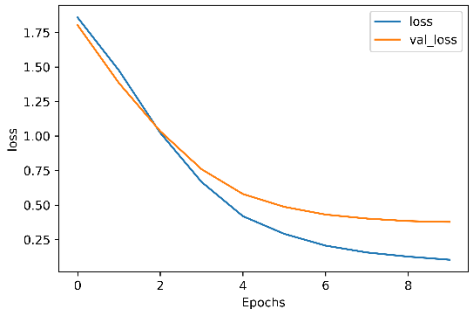

图 9： 损失值

可以看到在验证集上，大概在第 5 轮训练后模型的准确率趋于稳定值 89%，损失值也在不断下降，降到了 0.50 以下。

我们发现模型在训练集和验证集上的表现有些差距，这是过拟合的表现，为了解决这个问题可以进一步做数据集扩增、网络结构调整，超参数的调整等操作来让验证集上的表现更加接近训练集。

在数据不平衡的情况下，准确率往往不能很好的反应模型的表现，比如识别一个人是否身患某种疾病，患病的概率肯定要远小于未患病的概率，如果模型全部将结果判断为未患病，那么模型的准确率依旧可以很高，因此提出了 F1-score 指标:

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

它是查准率(precision)和查全率(recall)的调和平均数，查准率表示分类器判定正例中的正样本的比例，查全率则表示被预测为正例的占总的正例的比例。经过计算，我们的模型 F1-score 分数达到了 0.88，表现出色，可以很好的用于群众留言的自动分类。

3. 问题二的分析过程与结果展示

3.1. 总体流程图

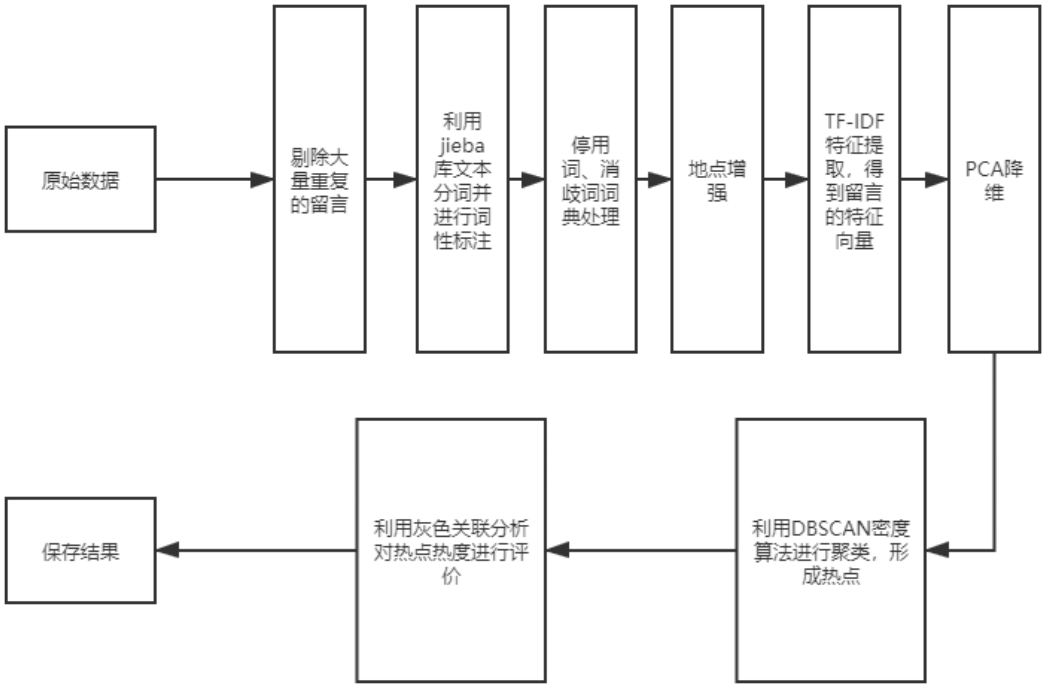

图 10： 总体流程图

3.2. 数据预处理

根据数据科学中“Garbage In， Garbage Out”的原理，对不满足数据质量要求的数据进行分析，往往得到的是不准确甚至错误的结论。这些“脏数据”将不同程度地影响数据挖掘的效率，因此数据清洗是整个数据挖掘过程中非常重要、不可缺少的一个环节，其结果质量直接关系到后续的数据挖掘分析以及最终结论。

因此，这一模块主要负责留言文本的数据预处理工作。在对已获得的留言数据进行热点话题发现之前，我们对用户留言的留言主题和留言详情进行如下操作：剔除大量重复的留言、利用 jieba 库文本分词并进行词性标注、停用词及消歧词词典处理、留言中的“地点”增强、TF-IDF 特征提取以及 PCA 降维。

3.2.1. 剔除大量重复的留言

我们观察发现，留言内容有的完全相同，有的部分重复。且针对同一问题，部分居民为了引起领导重视会进行重复留言。大量完全重复或部分重复的留言会影响我们的聚类与评价。因此我们需要对文本数据进行去重处理，去重即仅保留重复文本中的一条记录。

这里我们对留言的留言详情按逗号、句号等标点符号进行了语句分割，对4000 余条留言详情进行两两比对，如果语句的重复率高达 60%，且两条留言的留言用户若归属于同一人，则认为该留言是重复出现，无意义的。

经过去重处理后，最终剩下了 3313 条有效留言。

具体流程如下图所示：

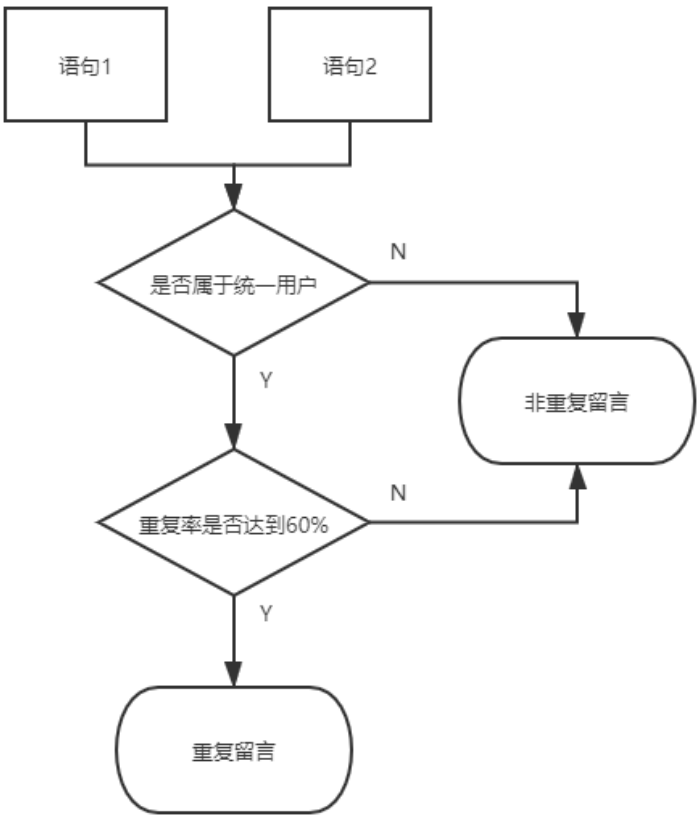

图 11： 去重流程图

3.2.2. 利用 jieba 库文本分词并进行词性标注

对留言详情进行去重处理后，我们留下了有一定价值的留言内容。为了便于后续的文本挖掘分析，我们需要把非结构化的文本信息转换为计算机能够识别的结构化信息。

在对群众留言进行分类时，我们已经预先对留言详情进行词语切分、未登

录词识别和词性标注的处理。

3.2.3. 停用词及消歧词词典处理

停用词是指在信息检索中，为了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，这些字或词即被称为 Stop Words（停用词）。停用词通常是人工输入、非自动化生成的，生成后的停用词会形成一个停用词表。为节省存储空间和提高搜索效率，我们需要对停用词进行处理。

对停用词进行过滤之后，我们还需要对歧义词进行筛选和处理。由于不同的语言单位，语义分析的任务各不相同。在词的层次上，语义分析的基本任务是进行词义消歧。

在对群众留言进行分类时，我们同样预先对留言详情进行停用词和消歧词的过滤。留下更有使用价值的用户留言，从而便于我们后续热点话题的挖掘。

3.2.4. 留言中的“地点”增强

留言中的“地点”增强，就是增加留言中地点词频的出现次数。因为对于聚类结果，我们更希望留言之间的相似性是更倾向于地点，而不是倾向于事件。比如以下两条留言：

A 市 107 国道东八线货车噪音扰民！

A5 区魅力之城小区一楼被搞成商业门面，噪音扰民严重

这两条留言描述的都是噪音扰民问题，这是它们的相似性；而两条留言一条属于 A 市，一条属于 A5 区，这是它们的差异性。我们不希望聚类时，算法将这两条留言归为同一类，所以我们需要增加表示地点的词频，增强留言间的差异度。具体操作流程如下所示：

（一） 留言的地点提取

留言主题中出现的地点大都是虚拟地点，因此尚未有现成的方法可以借鉴。这里我们观察了大量数据，总结出对于大部分留言主题，地点都会出现在开头的一般性规律，因此我们认为留言主题从前往后扫，市、县、区、乡、镇、村等表示地点的特征词最后出现的位置就是地点的结尾。如 A5 区劳动东路魅力之城小区临街门面烧烤夜宵摊，从留言的开头’A’开始，从前往后，遇到’区’但后面有’路’，所以继续扫，直到遇到’街’，’街’后面不再有地点特征词了，因此我们认为’街’就是地点的结尾，完整地点就是”A5 区劳动东路魅力之城

小区临街”。

当然还有部分留言并不是以地点开头，如：投诉 A 市伊景园滨河院捆绑销售车位，如果我们按照之前的方法，地点就会被标记为“投诉 A 市伊景园滨河院“，因此我们还需要对开头进行判断，如果开头的词是动词、动名词、副词等不相干的词汇，则跳过，这里的”投诉“是动词，所以我们跳过，地点会被正确标记为” A 市伊景园滨河院”。有了上述规律后，进行算法实现，结果如下：

图 12： 地点提取结果展示

可以看到，地点提取的效果还是非常不错的。

（二） 增加地点词频

目前已经得到了每条留言的地址，前文我们对留言也进行了 jieba 中文分词处理，因此这里只需要对留言的词语进行判断，如果词语属于地址的一部分，则增加该词语出现的次数。如 A2 区黄兴路步行街大古道巷住户卫生间粪便外排，留言的地址是“A2 区黄兴路步行街大古道巷“，留言的分词结果是[‘A2 区’,‘黄兴路’,’步行街’,’古道’,’住户’,’卫生间’,’粪便’,’外排’]，增加地点词频后的分词结果是[‘A2 区’, ‘A2 区’, ‘黄兴路’, ‘黄兴路’, ’步行街’, ’步行街’, ’古道’, ’古道’,’住户’,’卫生间’,’粪便’,’外排’]。

3.2.5. TF-IDF 特征提取

将用户留言进行去重、分词等预处理操作后，我们需要把这些词语转换为向量，便于后续留言文本的挖掘分析。我们选取 TF-IDF 算法，把用户留言信息最终转换为权重向量。

TF-IDF（term frequency–inverse document frequency）是一种用于信息检索与数据挖掘的常用加权技术。TF 指词频，IDF 指逆向文件频率。主要用于评估某个字词对于一篇文章或一个语料库里的一篇文章的重要性。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 加权的各种形式常被搜寻引擎应用，作为文件与用户查询之间相关程度的度量或评级。

TF-IDF 算法的主要思想是：如果某个词或短语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。

在本文中，我们将此思想应用于处理大量用户留言的这一基本环境，进行用户留言的特征提取并最终转化为特征向量。TF-IDF 算法的具体原理如下：

（一） 计算词频（TF）

词频 **(term frequency, TF)**，指的是某一个给定的词语在文本中出现的次数。由于同一个词语在长文本里可能会比短文本有更高的词频，而不管该词语重要与否。因此，词频通常会被归一化，以防止它偏向长的文本。

通常而言，词频（TF）= 某个词在留言文本中出现的次数

由于用户留言有长短之分，为了便于比较，我们将词频进行标准化处理。此时，

$$\text{词频（TF）} = \frac{\text{某个词在留言文本中出现的次数}}{\text{留言文本的总词数}} \qquad (3)$$

或者

$$\text{词频（TF）} = \frac{\text{某个词在留言文本中的出现次数}}{\text{该留言文本出现次数最多的词的出现次数}} \qquad (4)$$

（二） 计算逆向文件频率(IDF)

逆向文件频率 (inverse document frequency, IDF)是一个词语普遍重要性的度量。计算时，我们需要建立一个语料库用来模拟语言的使用环境。IDF 越大，此特征性在文本中的分布越集中，说明该分词在区分该文本内容属性的能力越

强。

由于一些通用的词语对于留言主题并没有起到太大作用, 反而一些出现频率较少的词能够表达留言的主题, 因此单一使用 TF 进行分析并不合适。在统计学上，权重的设计必须满足：一个词预测文本的能力越强，权重越大，反之，权重越小。因此在词频的基础上，要对每个词分配一个"重要性"权重。最常见的词给予最小的权重，较常见的词给予较小的权重，较少见的词则反而给予较大的权重。这个权重便是逆文档频率（IDF），它的大小与一个词的常见程度成反比。计算公式如下所示：

$$\text{逆文档频率（IDF）} = log\left(\frac{\text{语料库的留言文本总数}}{\text{包含该词的留言文本数}+1}\right) \quad (5)$$

（三） 计算 TF-IDF 值（TermFrequencyDocumentFrequency）

计算得到词频（TF）和逆文档频率（IDF）后，将这两个值相乘，就得到了一个词的 TF-IDF 值。如果某个词对文章的重要性越高，它的 TF-IDF 值就越大。因此，我们可以通过计算留言文本中每个词的 TF-IDF 值并进行排序，次数最多的即为初步提取的热点话题的关键词。它的计算如下所示：

$$\text{TF} - \text{IDF} = \text{词频（TF）} \times \text{逆文档频率（IDF）} \quad (6)$$

得到 TF-IDF 值后，我们便可以生成 TF-IDF 向量。它的具体实现步骤如下所示：

使用上述 TF-IDF 算法，找出每个留言主题中排名前 5 位的关键词

将提取出的这 5 个关键词合并为一个集合，计算每个留言主题对于这个集合的词的词频，如果没有则记为 0

生成各个留言主题的 TF-IDF 权重向量，如下所示：

$$\text{TF} - \text{IDF} = \text{词频（TF）} \times \text{逆文档频率（IDF）} \quad (7)$$

3.2.6. PCA 降维

在进行特征提取与处理时，涉及高维特征向量的问题往往容易陷入维度“灾难”。随着数据集维度的增加，算法学习需要的样本数量呈指数级增加。另外，随着维度的增加，数据的稀疏性会越来越高。在高维向量空间中探索同样的数据集比在同样稀疏的数据集中探索更加困难。

主成分分析（Principal components analysis，简称 PCA）是最重要的降维方法之一。在数据压缩消除冗余和数据噪音消除等领域都有广泛的应用。PCA 通常用于高维数据集的探索与可视化。还可以用于数据压缩，数据预处理等。PCA 可以把可能具有相关性的高维变量合成线性无关的低维变量，称为主成分（principal components）。新的低维数据集会尽可能的保留原始数据的变量。

假如留言文本数据集是 n 维的，共有 m 个留言数据。我们希望将这 m 个数据的维度从 n 维降到 n' 维，使之能够尽可能地代表原始数据集。基于最近重构性和最大可分性的思想，PCA 降维的算法流程如下所示：

输入：n维留言样本集 D=$(x(1), x(2),..., x(m))$以及要降维到的维数n'.

输出：降维后的留言样本集D''

对所有留言样本进行中心化：

$$x^{(\mathrm{i})} = x^{(\mathrm{i})} - \frac{1}{m}\sum_{j=1}^{m} x^{(j)} \qquad (8)$$

计算样本的协方差矩阵

对矩阵XXT进行特征值分解

取出最大的n'个特征值对应的特征向量$(w_1, w_2,..., w_{n'})$, 将所有的特征向量标准化后，组成特征向量矩阵w。

对样本集中的每一个样本$x(i)$,转化为新的样本

得到降维后的样本集D''

有时不指定降维后的n'的值，而是指定一个降维到的主成分比重阈值t。这个阈值t在（0,1]之间。假如我们的 n 个特征值满足$\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$,则$n'$可以通过下述条件计算得到:

$$\frac{\sum_{i=1}^{n'} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \geq t \qquad (9)$$

我们将留言主题经过 TF-IDF 算法进行特征提取后，每条留言特征向量的维度达到了 6593，这会为我们的计算训练时间带来极大的挑战。我们通过反复优化测试，最终通过 PCA 降维算法将数据维度降为 2000，累计贡献度为 90.26%，说明新的数据集尽可能的保留了原始数据的变量信息。

3.3. 综合评价

数据处理完毕后，我们将会使用 DBSCAN 聚类算法对各个相似留言进行聚

类，再定义留言总数、点赞数、点赞率、持续时间等指标，通过灰色关联分析等方法，得到各个聚类的热点指数，通过排序得到排名前 5 的热点问题，并进行分析。

3.3.1. DBSCAN 聚类算法

DBSCAN（具有噪声的基于密度的聚类方法）是一种基于密度的空间聚类算法。 该算法将具有足够密度的区域划分为簇，并在具有噪声的空间数据库中发现任意形状的簇，它将簇定义为密度相连的点的最大集合。因此，DBSCAN 聚类算法突破了传统聚类算法如系统聚类、K-means 聚类只能发现单一形状簇的局限性，在数据挖掘中具有更广泛的应用。

DBSCAN 算法的核心思想为寻找密度相连的最大集合，即从某个选定的核心对象（核心点）出发，不断向密度可达的区域扩张，从而得到一个包含核心对象和边界对象的最大化区域，区域中任意两点密度相连。

直观效果上看，DBSCAN 算法可以找到样本点的全部密集区域，并把这些密集区域当做一个一个的聚类簇。

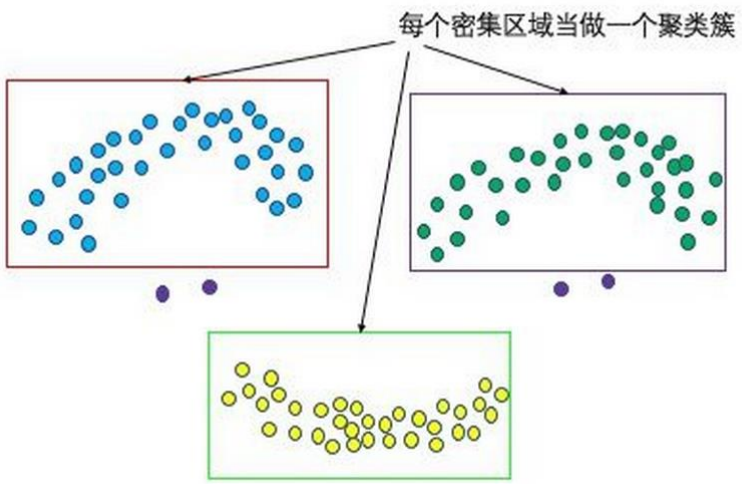

图 13： 聚类簇

我们选择两个参数，一个是正数 epsilon，它相当于指定的半径。另一个参数是自然数 minPoints，它是我们想要设定圈住的点的个数，相当于密度。以下图为例：

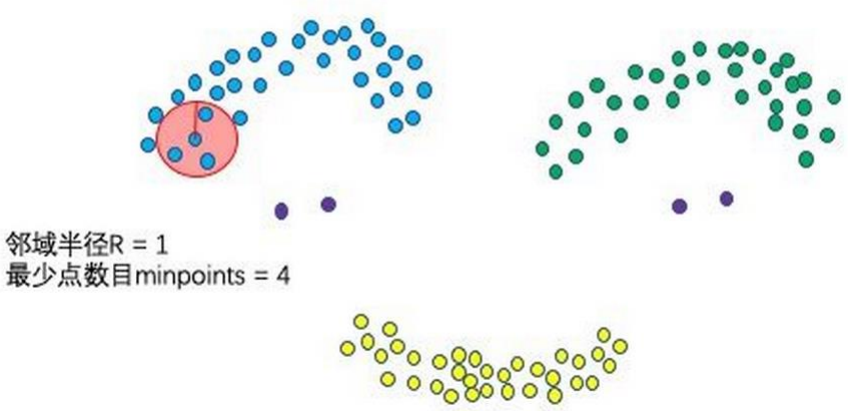

图 14： 两个参数

我们从数据集中选取某一任意点。如果从该点（包括原始点本身）到半径为 epsilon 的距离内有超过 minPoints 个点，我们则认为它们都是“集群”的一部分。然后，我们通过检查所有新点来扩展集群，并查看它们在半径为 epsilon 的距离内是否也有超过 minPoints 个点，如果是，则递归地增长集群。

我们将要添加到集群中的点一一进行了试验。最终选择一个新的任意点并重复这个过程。此时，这个点在它的以 epsilon 数值为半径的球中少于 minPoints 个点的情况是可能存在的，而且它也可能不是任何其他集团的一部分。如果是这种情况，它将被认为为独立的噪声点，也称离群点。

因此，邻域半径 epsilon 内样本点的数量大于等于 minpoints 的点叫做核心点。不属于核心点但在某个核心点的邻域内的点叫做边界点。既不是核心点也不是边界点的是噪声点。如下图所示：

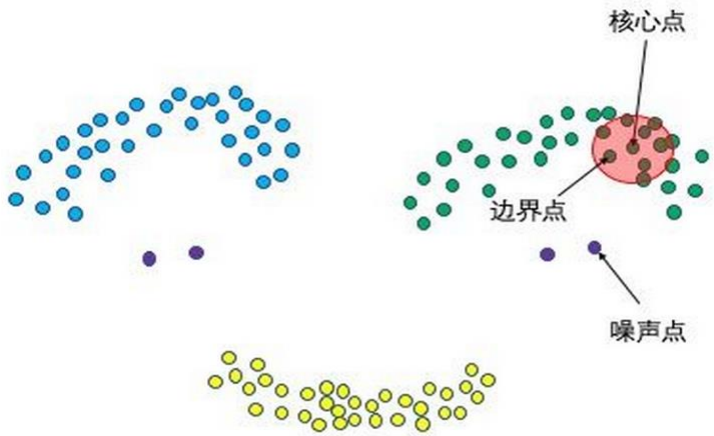

图 15： 三个点

DBSCAN 的算法步骤可以分成两步。

1、寻找核心点形成临时聚类簇。

我们扫描全部样本点，如果某个样本点，它半径范围内点的数目≥ minpoints 个，则将其纳入核心点列表，并将其密度直达的点形成对应的临时聚类簇。

2、合并临时聚类簇得到聚类簇。

对于每一个临时聚类簇，检查其中的点是否为核心点，如果是，将该点对应的临时聚类簇和当前临时聚类簇合并，得到新的临时聚类簇。重复此操作，直到当前临时聚类簇中的每一个点要么不在核心点列表，要么其密度直达的点都已经在该临时聚类簇，该临时聚类簇升级成为聚类簇。继续对剩余的临时聚类簇进行相同的合并操作，直到全部临时聚类簇被处理。如下图所示：

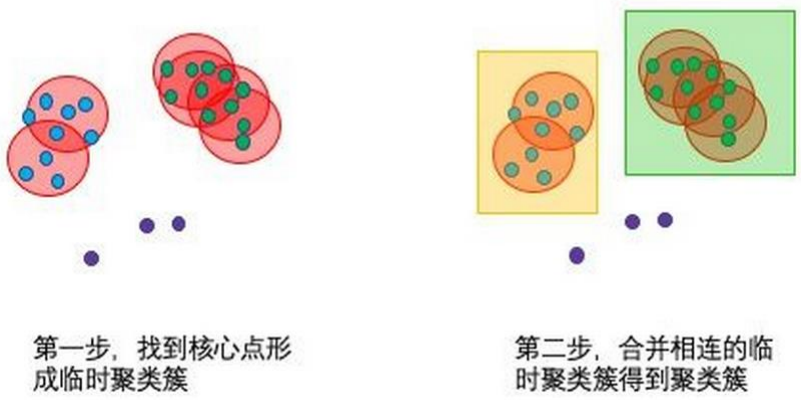

图 16： 两步算法

经过对参数不断进行调试并实践，我们将 epsilon 设定为 0.86，minpoints 设定为 2，进行 DBSCAN 聚类分析。最终共生成了 409 个类和 2132 个离群点。通过相关指标建立和灰色关联分析，我们将筛选出热度排名前五位的热点话题。

3.3.2. 评价指标选择

考虑我们之后会用上的综合指标模型方法，所以我们在指标选择时会尽可能选择效应型指标，让所有指标对最终评价得分的结果影响都是正向的。

1、留言个数

这里的“留言个数”指的是一个热门问题所涉及的所有留言的总数，也就是有多少人针对这个问题进行了留言，直接表现了该问题的被关注程度。留言数越多自然其被关注度越高，那么它最终评价的得分应该相应更高，所以该指标符合了效应型指标的特征。

2、持续时间

“持续时间”指的是一个热门问题在所给数据中持续存在的时间长度，其计算公式如下所示：

持续时间=问题涉及的最近的留言时间-问题涉及的最近留言时间（10）

简而言之“持续时间”就是热门问题涉及的最近一次的留言时间与最早的留言时间的差值，该指标表明该热门问题持续存在而未被解决时间范围长度。该问题的“持续时间”越长，人们对它的关注度也会越大，那么它的最终评价的得分也相应更高，所以该指标符合了效应型指标的特征。

3、总赞成数与赞成率

“总赞成数”指的是一个热门问题所涉及的所有留言的赞成数的总和，计算公式如下所示：

$$\text{总赞成数} = \sum \text{问题所涉及的留言的赞成数} \quad (11)$$

但考虑到每个留言还有反对数的存在，所以有“总赞成数”这个指标是不够的，并且因为我们尽可能寻找的是效应型指标，所以我们再引进“赞成率”这个变量，其计算公式如下所示：

$$\text{赞成率} = \frac{\text{总赞成数}}{\text{总赞成数+总反对数}} \quad (12)$$

其中：

$$\text{总反对数} = \sum \text{问题所涉及的留言 的赞成数} \quad (13)$$

“赞成数”和“赞成率”的大小都与热门问题的被关注度成正相关，所以均符合了效应型指标的特征。

3.3.3. 灰色关联分析

灰色关联分析是分析系统中各因素关联程度的一种方法，通过计算参考列表和比较列表的差异性来判断数值的关联程度，即关联度。关联度越大，说明数值与参考列表越一致，该指标在整个指标体系中重要程度就越大，权重也就越大。具体步骤如下：

（1）确定比较对象（评价对象）和参考数列（评价标准）。设评价对象有 m 个，评价指标有 n 个，参考数列为$x_0 = \{x_0(k)|k = 1,2,\cdots,n\}$，比较数列为$x_i = \{x_i(k)|1,2,\cdots,n\}$，$i = 1,2,\cdots,m$。

（2）计算关联系数和关联度：

$$\varepsilon_{0i}(k) = \frac{\min\limits_s \min\limits_t |x_0(k)-x_s(k)| + \rho \max\limits_s \max\limits_t |x_0(k)-x_s(k)|}{|x_0(k)-x_i(k)| + \rho \max\limits_s \max\limits_t |x_0(k)-x_s(k)|} \quad (14)$$

为比较数列x_i对参考数列x_0在第 k 个指标上的关联系数，其中$\rho \in [0,1]$为分辨系数，我们这里取$\rho = 0.5$。其中，称$\min\limits_s \min\limits_t |x_0(k) - x_s(k)|$、$\max\limits_s \max\limits_t |x_0(k) - x_s(k)|$分别为两级最小差及两级最大差。

一般来讲，分辨系数ρ越大，分辨率越大；ρ越小，分辨率越小。关联度的计算：

$$r_{0i} = \frac{1}{n}\sum_{k=1}^{n} \varepsilon_{0i}(k) \quad (15)$$

各个数列的关联度大小，直接反映了各个评价指标相对于设定数列的相对重要的程度。

（3）以r_{0i}作为各个评价指标的权重值，及$w_i = r_{0i}$。

通过计算我们得到四个评价指标：持续时间、点赞总数、点赞率、留言总数的权重分别是 0.24122749, 0.18013493, 0.38790072, 0.19073687，说明问题的点赞率和留言总数对于问题是否是热点问题影响较大，问题如果点赞率高、留言数量大，那么该问题就极有可能被认为是热点问题。

（一）热点指数计算

在将各评价指标用于计算热点指数前我们还需要消除量纲的影响，进行归一化：

$$Z_{ij} = \frac{X_{ij} - X_{j_{min}}}{X_{j_{max}} - X_{j_{min}}} \quad (16)$$

这样所有的数据都会分布在[0,1]之间，成为标量。于是第i条热点的热点指数计算公式为：

$$Score_i = \sum_{j=1}^{4} w_j \times Z_{ij} \quad (17)$$

再对热点指数进行排序，我们就可以得到最热门的前五个问题，接下来我们还需要从它的留言详情和留言主题信息中筛选出其地区/人群的数据，因为在n.3.1 中通过利用留言详情中关键词进行的聚类应该已经对地区/人群有了初步的聚类，所以我们可直接对五个问题中的留言详情和留言主题里的地区/人群数据进行人工筛选，聚类后已经让此工作变得十分简便，结果如下：

表 6： 热点问题按照地点/人群划分

热点排名	问题 ID	热度指数	时间范围	地点/人群	问题描述
1	1	0.583828245	2019/4/2 至 2019/12/15	A 市	关于 A 市人才购房补贴的疑问
2	2	0.422779748	2019/1/15 至 2019/4/3	A2 区丽发新城	投诉 A2 区丽发新城附近建搅拌站噪音扰民
3	3	0.42215979	2019/10/21 至 2019/12/23	A 市伊景园滨河苑	A 市伊景园滨河苑捆绑销售车位
4	4	0.411568563	2019/1/22 至 2019/5/31	西地省科技职业学院	西地省科技职业学院宿舍出现种种问题
5	5	0.389700052	2019/7/15 至	A 市五矿万境	A 市五矿

			2019/8/1	K9 县	万境 K9 县房屋出现质量问题

由结果可以看到发生 A 市的“关于 A 市人才购房补贴的疑问”问题成为了最热问题，热度指数高达 0.584，有关部门应该及时就“A 市人才购房补贴”进行针对性的回复或解决。

3.3.4. 结果分析

前五热门度的“热门问题“的具体指标数据如下：

表 7： 热点问题指标数值

热度排名	持续时间（天）	点赞数（个）	点赞率（%）	留言总数（条）
1	343	27	0.931034	31
2	137	2106	1	6
3	354	32	1	6
4	348	9	1	5
5	355	3	1	4

可以看出，“热门问题”的点赞率比较一致地接近或等于 1，所以点赞率对问题的综合评价的影响最小；而除了热度第一的问题外，留言总数也几乎相等，热度第一的问题因其较多的留言总数评价靠前，留言总数对综合评价的影响最大；第二热度问题则有超高的点赞数，相比于持续时间更长的第三热度问题其综合评价更靠前，所以点赞数对综合评价的影响更大，持续时间次之。综上所述，影响“热门问题”的评价的主要指标影响力大小：留言总数>点赞数>持续时间>点赞率。

4. 第三问的分析过程和结果展示

4.1. 相关性

4.1.1. 数据预处理

在正式对答复意见进行评价之前，我们需要先对留言详情及答复意见进行数据预处理操作，以保证后续对留言详情与答复意见的相关性、完整性和可解释性的分析与评价过程更合理、准确。

（一） 利用 jieba 进行中文分词

为了对留言详情和答复意见进行三个性质的评价，我们首先需要把非结构化的文本信息转换为计算机能够识别的结构化信息，这是后续操作的基础。通过 python 导入 jieba 库并进行词语切分、未登录词识别等操作，我们初步对留言详情和答复意见进行了中文分词处理。

（二） 停用词、消歧词词典处理

停用词指没有实际含义、不作为结果的词，如冠词、介词、副词或连词等。此外，中文分词器切分出来的所有词条中含有大量的单个独立字，它们均降低了分类系统的处理效率和准确度。因此，我们在分词后需要对留言详情和它们的答复意见进行停用词词典处理。对停用词进行过滤后，我们还需要对消歧词进行筛选和处理。上述操作可以进一步对留言详情和答复意见实现文本的数据预处理。

（三） TF-IDF 特征提取，得到留言的特征向量

将留言详情和答复意见进行中文分词、停用词及消歧词词典处理等预处理操作后，我们需要把这些词语转换为向量，便于后续留言文本的挖掘分析。我们选取 TF-IDF 算法，通过分别计算词频（TF）、逆向文件频率(IDF)以及它们的乘积 TF-IDF 值后，我们把用户留言信息最终转换为它的特征向量。

4.1.2. 相关性计算

为了对答复意见与留言详情的相关性程度进行评价，我们利用相似度度量来对文本之间的相关性进行等价和量化。

相似度度量（Similarity），即计算个体间的相似程度，相似度度量的值越小，说明个体间相似度越小，相似度的值越大说明个体差异越大。

对于多个不同的文本或者短文本对话消息要来计算他们之间的相似度大小，一个好的做法便是将这些文本中的词语映射到向量空间，从而形成文本中

文字和向量数据的映射关系，通过计算几个或者多个不同的向量的差异大小，来计算文本之间的相似度。

在机器学习算法中，计算对象间距离的方法有很多种。基于词向量而言，可以使用的距离度量有曼哈顿距离、欧几里得距离等。在本研究中，我们选择了对于衡量对象间相似度更为成熟合理的余弦相似度。余弦相似度是用向量空间中两个向量夹角的余弦值作为衡量两个个体间差异的大小的度量，取值范围为[-1,1]，余弦值接近 1，夹角趋于 0，表明两个向量越相似；余弦值接近于 0，夹角趋于 90 度，表明两个向量越不相似。因此，值的大小可以衡量向量之间的相似度。相比其他的距离度量方法，余弦相似度更加注重两个向量在方向上的差异，而非仅仅在于距离或长度。因此，在本题中，我们认为相关性是计算“留言详情”与“答复意见”特征向量之间的余弦值，取值在 0-1 之间。

计算对象间的余弦相似度的流程如下所示：

（一） 将留言详情与答复意见进行中文分词

我们以某条留言详情和答复意见举例说明。将留言详情定为句子 A，答复意见定为句子 B。

句子 A: 建议将“白竹坡路口”更名为“马坡岭小学”，原“马坡岭小学”取消，保留“马坡岭”

句子 B：您好！您的留言已收悉。现将有关情况回复如下：关于来信人建议“白竹坡路口”更名为“马坡岭小学”，原“马坡岭小学”取消，保留“马坡岭”的问题。公交站点的设置需要方便周边的市民出行，现有公交线路均使用该三处公交站站名，市民均已熟知，因此不宜变更。感谢来信人对我市公共交通的支持与关心！

在数据预处理阶段，我们已经将留言详情和答复意见进行 jieba 中文分词，它们会以列表的形式分别存放于 listA 和 listB 中。

（二） 将分词合并成一个集合

将 listA 和 listB 合并为一个集合，集合中汇聚的是在两个句子中出现且不重复的词语，便于我们观察与分析。

（三） 进行词频向量化

将句子 A 和句子 B 的词频分别进行统计后，我们列出两个句子的词频向量，方便后续余弦相似度的计算。

句子 A：（0，0，1，0，0，0，0，0，0，0，0，0，0，0，0，1，0，3，0，0，0，0，0，4，0，0，0，0，0，0，2，0，0，0，1，1，0，0，1，2，1，

0，0，0，0，4，0，0，0，0，0，1，0，0，0，1，1，0）

句子 B:（1，1，1，1，1，1，1，3，1，1，1，1，1，1，2，1，1，3，1，2，1，1，2，4，1，1，1，1，2，1，5，1，1，2，1，0，1，1，1，2，1，2，1，1，1，1，1，5，1，1，1，1，1，1，1，1，1）

（四） 用余弦函数计量两个句子的相似度

通过向量表示出两个句子的词频向量，便可以利用公式计算两个向量之间夹角的余弦值，且值越大表示向量间的相似度越高，即相关性越大。具体公式如下所示：

$$\cos\theta = \frac{\sum_{i=1}^{n}(x_i \times y_i)}{\sqrt{\sum_{i=1}^{n}(x_i)^2} \times \sqrt{\sum_{i=1}^{n}(y_i)^2}} \quad (18)$$

通过 python 导入 jieba 库、numpy 库和 re 库以及相关操作，最终我们得到列举出的留言详情以及它对应的答复意见的余弦相似度 0.6307573635273277。数值大小中等偏上，说明此条留言详情和它的答复意见的相似度大体可观，相关性良好。

4.2. 完整性

用主谓宾来衡量句子完整性，整个答复的完整性就是回答中所有句子完整性的平均值。

4.2.1. 数据清洗

将答复以“。”、“!”、“？ ”分割成一个个句子。再进行 jieba 分词，但是不进行停用词、消歧词词典处理，因为句子中的每个成分都是我们分析句子主谓宾结构的依据。

4.2.2. 完整性计算

我们对完整性的定义主要为句子的完整性，一般完整的句式主要有主语、谓语、宾语三个成分，缺一不可。所以完整性可以根据三个成分的在句子中的出现情况计算，计算思路为句子实际成分总数与句子应有成分总数的比值。主要有以下四种情况，分别对应四种公式：

1.没有主语：（谓语数+宾语数）/（1+谓语数+宾语数）

主语在句子中必须存在，而且主语通常是可以一对多或多对一的分别对应谓语或宾语，所以在主语不存在的局势中我们可以当作主语应有数为 1。

2.没有谓语：（主语数+宾语数）/（1+主语数+宾语数）

谓语和主语或宾语通常都是一对多的关系，所以在谓语不存在的句子中我们可以当作谓语应有数为 1。

3.缺少宾语：（主语数+谓语数+宾语数）/（主语数+谓语数*2）

缺少宾语主要指的是宾语数<谓语数的情况，因为谓语与宾语之间只有一对多的关系，所以句子中应有的宾语数应该同句子中谓语数相等。

4.句子完整没有任何成分缺失：1

（对于主语、谓语、宾语中没有任何一个的句子可以忽略，不影响整个回复的句子完整性）

问题就转变成了计算句子中主语、谓语、宾语出现的次数，而目前的 NLP 方法中未有能直接计算句子中的主谓宾数量的。因为句法分析是通过句子中语言单位内成分之间的关系来对词语语法功能进行分析，揭示其句法结构。所以我们想到了利用对句子的语法、依存句法分析来完成主谓宾数量的统计。

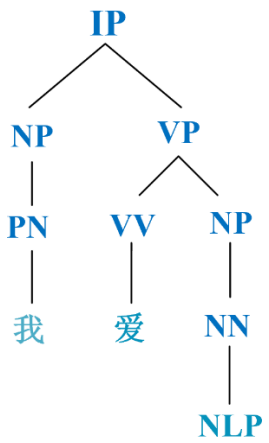

图 17： 句法分析

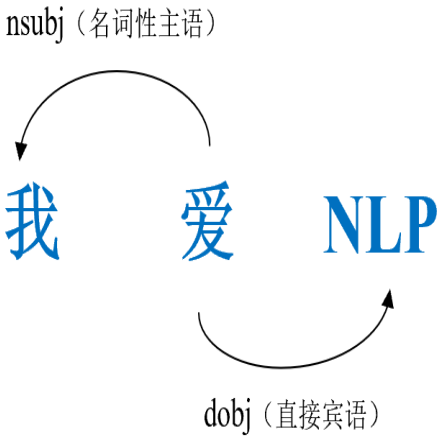

图 18: 主谓宾切分

目前已有复旦大学 fnlp、Stanford NLP、Hanlp、哈工大 ltp 等多种句法分析工具，其中以 Stanford NLP 的名声最为之盛，本文我们将使用 Stanford NLP 来进行答复的完整性分析。

Stanford NLP 是由斯坦福大学的 NLP 、小组开源的 Java 实现的 NLP 工

具包，能提供分词、词性标注、命名实体识别、句法分析，依存句法分析等功能，句法分析，依存句法分析使用方便，只需在斯坦福 nlp 官网下载 stanford-parser-full-2018-10-17 工具包，里面会为我们提供现成的分析工具，我们可以直接调用从而完成句法分析。

得到两两分词之间的依存关系后，我们就可以计算句子中主谓宾出现的次数，计算规则如下：

表 8： 计算规则

依存关系	结构	计算规则
Dobj（直接宾语）	动词 --> 直接宾语	谓语、宾语数+1
Pobj（介宾）	介词 --> 宾语	宾语数+1
Iobj（间接宾语）	动词 --> 间接宾语	宾语数+1
Csubj（主语从句）	谓语动词 --> 主语从句中的主要成分	谓语数+1
Csubjpass（主从被动）	谓语动词（被动）--> 主语从句中的主要成分	谓语数+1
Nsubj（名词性主语）	句子的主要成分（一般是动词）--> 主语	主语数+1
Nsubjpass（被动名词主语）	句子的主要成分 --> 主语（被动）	主语数+1

这样我们通过遍历回答中的每个句子，计算句子中的主谓宾数量，就可以得到句子的完整性，随后计算出整个回复所有句子的平均完整性，作为整个回复的完整性。

4.3. 可解释性

我们答复的可解释性表示能否让群众信服，是否有说服力。因此我们将该指标定义为回复中是否含有法律法规。因为答复者帮助提问者解决问题如果是依据法律执行的，一定会更有信服力。其计算公式如下：

$$可解释性=\begin{cases}1，答复中引用了法律法规\\0，答复中未引用法律法规\end{cases} \quad (19)$$

通过观察数据发现，法律法规都会被“《”，“》”符号包围，因此我们认定如果答复中出现了符号“《”，“》”则认为回复者引用了法律法规，但是仍存在少部分数据如：网友“A000110735”：您好！您在平台《问政西地省》上的留言已收悉…这些数据会存在偏差，因为我们错误的把“问政西地省”也当作了法律法规，事实上它只是一个留言平台，不具有可解释性。因此我们还需要加入判断条件，如果符号“《”，“》”内的是“问政西地省”，则不算作引用了法律法规。尽管这样做无法顾及到所有的数据，但是该方法对于大部分数据却是完全准确的，相比于我们训练 word2vec 模型来区分哪些是法律法规来说，显得更加准确、高效。

4.4. 结果展示

通过上述我们对“相关性”，“完整性”，“可解释性”这三个主要指标的定义，我们计算出了各个回复的三个指标的数值，部分结果展示如下：

表 9： 部分答复意见的评价

留言编号	留言用户	留言详情	答复意见	完整性	相关性	可解释性
3681	UU00812	我做为一东澜湾社区居民,我替社区几千户居民为民请愿,小区小孩子老人特别多…	网友“UU00812”您好…2019 年 1 月 15 日	0.111111	0.218898	1
4111	UU00811139	尊敬的 A 市市委领导，您好！A3 区银杉路 559 号岳银欣苑小区停车问题反映如	网友“UU001139”您好…2018 年 12 月 4 日	0.159722	0.637404	0.5

		下…				
4331	UU0082338	尊敬的书记：您好！经过查看在政府备案的该楼盘的装修价格的评估报告分析，我们业主…	网友“UU0082338”您好…2018年11月2日	0.139365	0.519159	1
4824	UU008880	A市近年来经济发展取得长足的进步，各项经济指标在全国也是靠前的。但是，生活中的小事…	网友“UU008880”您好！您的留言已收悉…2018年10月17日	0.066667	0.359864	1
4858	UU008653	尊敬的书记：您好！经过上次恒大大绿洲业主留言幼儿园园长收贿赂导致很多业主子女无法上学…	网友“UU008653”您好！您的留言已收悉…2018年9月7日	0.055556	0.352327	0.5
5088	UU0082034	https://baidu.com/.20开盘，当天售罄，经过近三个月的漫长等待迎来网签。但看到合同的时候大多数人都有意见…	网友“UU0082034”您好！您的留言已收悉…2018年9月30日	0.207738	0.198574	1
5246	UU0082	坚决反对修改	网友	0.110	0.281	1

	2	乾城二期原规划，理由如下：1、乾城二期是一个整体，于2015年已经开发并交房了7栋…	“UU00822”您好！您的留言已收悉…2018年8月10日	043	736	
5328	UU008989	我家住A6区月亮岛街道黄狮岭社区时代倾城小区，目前该小区属于全区最大的小区，但是配套…	网友“UU008989”您好！您的留言已收悉…2018年8月17日	0.122024	0.33896	1
5399	UU00815	小区入住已有十多年，至今未成立业委会，社区对几个热心牵头成立业委会的居民不闻不问，忽悠…	网友“UU00815”您好！您的留言已收悉…2018年7月23日	0.111706	0.513702	0.5
5420	UU008404	胡书记：您好！本人是北辰三角洲小区的业主，小孩到了上幼儿园的年纪，户口已迁入北辰…	网友“UU008404”您好！您的留言已收悉…2018年7月10日	0.104938	0.575361	1

4.5. 各指标数值分析

（一）完整性

根据示例结果数据可知，完整性的计算结果数值较为稳定，且数值偏小，对回复整体的评价的贡献率较低。

（二）相关性

相关性的计算结果数值则不同于完整性，其波动范围较大，方差较大，能表现各个回复整体评价间的大部分差异，所以相关性对回复的整体评价的贡献率较高。

（三）可解释性

因为可解释性属于二项分布指标，可解释性对回复整体评价的造成的偏差时较大的，其计算结果的两个数值频率如下：

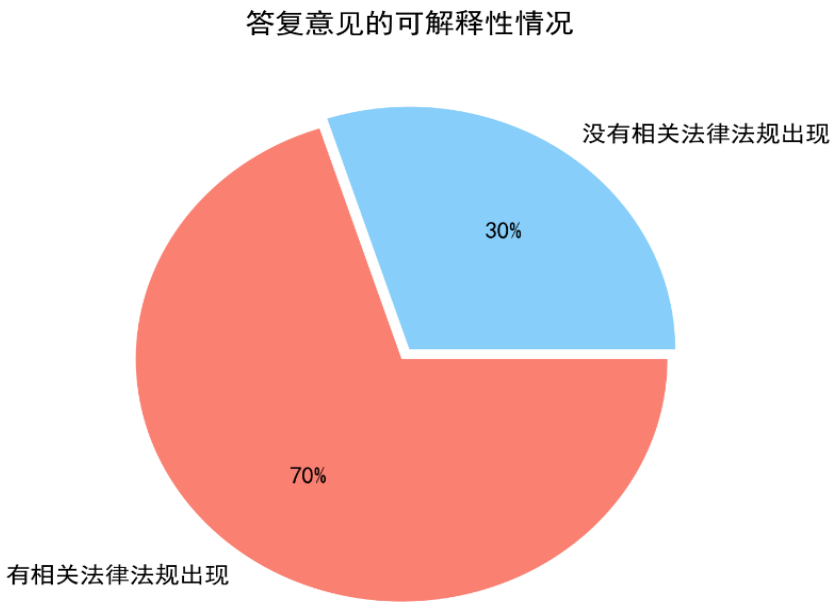

图 19： 可解释性比例图

由图可以看出，有相关法律法规出现的留言是占了大部分比例的，价值我们认为，对一个留言的评价，具有较好评价的留言里有相关法律法规的出现作支撑是必不可少的，这表现的是一个留言的可靠性与可信性。而有相关法律法规出现的留言比例占了大部分，所以在这些留言中找出较好评价的留言，可解

释性造成的偏差应该是较小的，所以若使用可解释性对评价较好的留言的筛选是非常可行的。

5. 参考文献

[1] 郝媛媛, 叶强, 李一军. 基于影评数据的在线评论有用性影响因素研究[J] 管理科学学报, 2010, 13(8): 78-88.

[2] 王娟, 慈林林, 姚泽康. 特征选择方法综述.计算机工程与科学. 2005,(12):68-71

[3] 刘龙飞, 杨亮, 张绍武,等. 基于卷积神经网络的微博情感倾向性分析[J]. 中文信息学报, 2015, 29(6): 159-165.

[4] 吴庆涛，普杰信，崔林. 基于 BBS 文本信息的数据挖掘.洛阳工学院学报，2002，(2):55-58.

[5] 周茜,赵明生等.中文文本分类中的特征选择研究[J].中文信息学报. 2003, 18(3): 17-23.

[6] 邱立坤，程葳，龙志炜，等. 面向 BBS 的话题挖掘初探[C]//全国第八届计算语言学联合学术会文集. 南京. 2005：401-407.

[7] 骆卫华，刘群，程学旗.话题检测与跟踪技术的发展与研究，全国计算语言学联合学术会议（JSCL）， 2003.

[8] 宋丹，卫东，陈英.基于改进向量空间模型额度话题识别跟踪.计算机与发展，2006,9(16):62-67.

[9] 李昕，朱永盛，武港山.论坛帖子语义结构的提取与分析.第一届全国信息检索与内容安全学术会议，上海，2004：172-179.

[10]何国斌，赵晶璐.汉语文本自动分词算法的研究，2010,(3):125-127