基于机器学习的留言数据挖掘与模型构建

摘要

近些年来，随着互联网的飞速发展，微信、市长信箱、阳光热线等网络平台逐步成为了政府了解民意、凝聚民气以及汇聚明智的主要渠道，各类社情民意相关的文本数据量不断攀升。为了提高政府的管理水平以及施政效率，本文将建立基于自然语言处理技术以及文本挖掘的方法，建立留言内容的分类模型，归类热点问题以及结合量化与质化数据分析，探讨出一套合理的评价方案。

针对问题一：首先，我们对文本数据进行预处理，根据定义相应的规则剔除掉毫无意义的留言内容，将文本数据处理成与分类模型相对应的格式并训练分类模型。本文采用了支持向量机模型，将文本通过计算抽象成向量化的文本数据，提高分类精确度。再使用卡方统计量进行降维处理，将所有类别的 N 个特征向量进行合并得到最终的特征向量。根据最终的特征向量输出所有文本的量化格式并对分类模型进行 knn 算法的训练和预测。

针对问题二：该题基于 LDA 模型的社会化标签综合聚类方法,通过建立相似标签处理相似的留言内容，将特定地点或人群的数据归并，同时将相似的留言内容归于一类。为了更好地表达留言内容的语义信息，我们建立词对向量模型并结合主题模型，挖掘出留言文本中的潜在语义，把两个模型得到的特征进行特征融合，并应用 K-means 聚类算法进行留言内容聚类和 lda 困惑度计算，得出最优主题，并通过 hadoop 体系架构对数据的一个分析，得出热点问题的排名。

针对问题三：我们首先对政府答复的相关文本数据进行量化处理，根据余弦相似度与相似系数为指标计算留言详情和答复意见的相关性；其次，建立答复规范模式结合相关性对答复意见的完整性进行评价；最后结合结合量化与质化数据分析以及可解释性的关系建立出一套评价方案。

关键词： 特征提取；LDA 模型；K-means 聚类；knn 算法；Rnn 神经网络

Abstract

In recent years, with the rapid development of the Internet, WeChat, mayor's mailbox, sunshine hotline and other network platforms have gradually become the main channels for the government to understand public opinion, build up popular sentiment and gather wisdom, and the amount of text data related to various social situations and public opinions keeps increasing.In order to improve the management level and administrative efficiency of the government, this paper will establish a method based on natural language processing technology and text mining, establish a classification model of message content, classify hot issues and combine quantitative and qualitative data analysis to explore a reasonable evaluation program.

Aiming at task one,first, we preprocessed the text data, eliminated the meaningless message contents according to the corresponding rules defined, processed the text data into the format corresponding to the classification model and trained the classification model.In this paper, a support vector machine (SVM) model is used to abstract text into vectorized text data by computing.Then, chi-square statistics is used for dimensionality reduction, and N eigenvectors of all categories are combined to obtain the final eigenvectors.According to the final feature vector output all text quantization format and classification model KNN algorithm training and prediction.

Aiming at task two, the LDA model-based integrated clustering method of social tags is used to establish similar tags to deal with similar message contents, merge data of specific places or people, and classify similar message contents into one category.In order to better express the semantic information of contents of a message, we have set up a word on vector model and combining the topic model, dig out the potential meaning of the text message, the two models have the features of integration, and application of K - means clustering algorithm to the message content and lda confused degree calculation, it is concluded that the optimal theme, and through the hadoop architecture an analysis of the data, draw a hot issue.

Aiming at task three, we first conducted quantitative processing on the text data related to the government's reply, and calculated the correlation between the message details and the reply comments based on the cosine similarity and similarity coefficient.Secondly, a reply standard model is established to evaluate the integrity of the reply with relevance.Finally, a set of evaluation scheme is established based on the relationship between quantitative and qualitative data analysis and interpretability.

Key words: feature extraction; LDA model; k-means clustering; KNN algorithm; Rnn neural network;

目录

一、问题分析

随着互联网时代的到来，微信、微博、市长信箱、阳光热线等网络问政平台逐步成为了政府了解民意、汇聚明智、凝聚民气的重要渠道。各类社情名意的文本数据不断攀升，给有关部门的工作处理带来了极大的挑战。对于民众的留言内容，相关部门的工作人员需要进行人工的对文本数据进行归类、划分、整理，需要花费大量的时间、物力以及人力。因此随着大数据、云计算、人工智能等技术的发展，我们可以利用自然语言语言处理以及文本挖掘的方法，对群众留言进行归类，挖掘热点以及对答复意见进行评价，提高部门的工作效率。

问题给出了四个附件的数据，附件 1 提供了对留言内容分类的三级标签体系，这三个标签体系相互联系；附件 2 主要包括了群众的留言内容，时间以及留言主题；附件 3 包括了群众的留言时间、留言详情、留言主题以及反对数和点赞数；附件 4 除了群众的留言详情之外，还给出了相关部门的答复意见。

对于问题一，主要是要求我们根据附件 2 提供的文本数据，对数据进行文本处理，将文本数据转化为词向量模型，利用 RNN 循环神经网络，从而建立关于留言内容的一级标签分类模型。

对于问题二，要求我们根据附件 3 给出的数据，对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，并给出评价结果，并按照问题提供的表一以及表二模板，解决相关问题；

对于问题三，我们需要根据附件 4 相关部门对留言的答复意见，从答复的相关性、完整性、可解释性等角度对答复意见的质量给出一套评价方案。

二、数据处理

2.1 剔除不需要的文本

根剧附件二群众的留言详情可以发现，有些留言详情会对留言内容分类造成干扰，留言内容与文本提取没有直接关系，这样会对研究问题造成一定难度，故删除。

2.2 删除与分析无关的指标

通过问题分析明确所需要的指标，基于准确、有效、易量化的原则对给出指标进行有选择的保留。最终保留的指标有留言详情、留言主题、分类标签、答复意见、留言编号、留言时间以及反对数、点赞数。

2.3 构造分析需要的指标

（1）词向量

为了更好地实现对评论文本进行聚类，本模型将主题信息融合到评论文本词向量训练的过程。使用 LDA 获得该条评论的主题信息，和原有的评论内容进行拼接，作为评论与主题信息结合后的向量表达。将前述得到的融合主题信息的评论文本作为输入，训练 CBOW 模型。

（2）TF-IDF 矩阵

TF-IDF(Term Frequency-Inverse Document Frequency)，中文叫做词频－逆文档频率。在文本挖掘(Text Mining)和信息检索(Information Retrieval)领域具有广泛的应用。TF-IDF 通过计算每个词的 TF-IDF 值，筛选出每个文档中最关键一些词。

TF-IDF（Term Frequency－Inverse Document Frequency），就是平衡这两者的产物，它由两个部分相乘得到：TF × IDF。其中

$$TF=\frac{某词文档中出现的次数}{文档的总词数}$$

TF 值越大，词的存在感越强，他是将特征 1 进行量化

$$IDF=\log(\frac{语料库的文档总数}{包含改词的文档数+1})$$

语料库的作用在于模拟某种语境，当 IDF 值越大，说明在语境中包含该词的文档树越少，这个词越具有唯一性代表性，某种意义上说，它越关键。它是将特征 2 进行量化。

（3）lda 困惑度

对于 LDA 模型，最常用的两个评价方法困惑度（Perplexity）、相似度（Corre）。其中困惑度可以理解为对于任意一条留言内容，所训练出来的模型对语料库属于哪个主题有多不确定，这个不确定成都就是困惑度。困惑度越低，说明聚类的效果越好。

计算公式为：

$$Perplexity（D）=exp（-\frac{\sum \log P(w)}{\sum_{d=1}^{M} N_d}）$$

其中分母是测试集中所有单词之和，即测试集的总长度，不用排重。其中 p(w)指的是测试集中每个单词出现的概率，计算公式如下：

$$P(w)=p（z|d）*p(w|z)$$

其中 p(z|d)表示的是一个文档中每个主题出现的概率，p(w|z)表示的是词典中的每一个单词在某个主题下出现的概率。对于不同 Topic 所训练出来的模型，计算它的困惑度。最小困惑度所对应的 Topic 就是最优的主题数。

（4）问题热点排名

在处理热点排名问题，运用 hadoop 体系的 mapreduce 框架处理排名问题，效率较快，原理如下图：

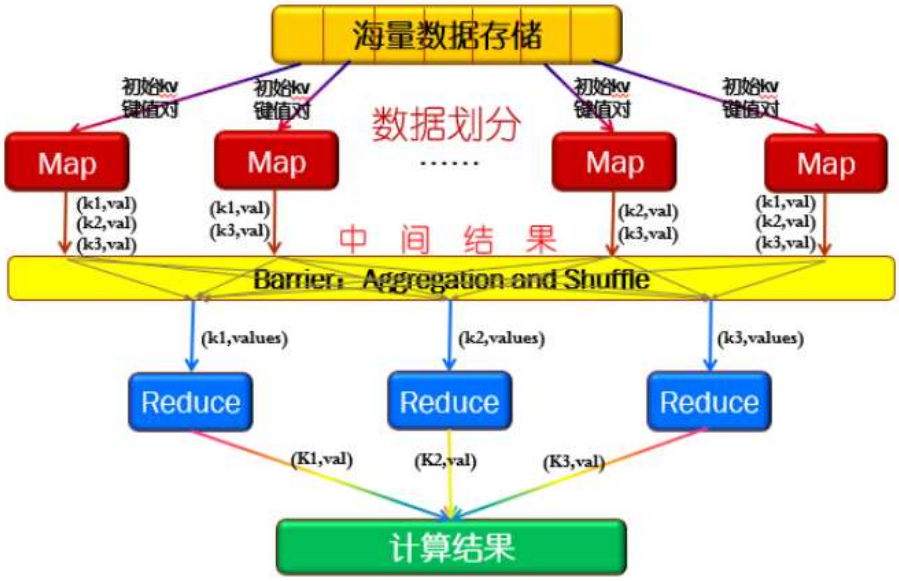

图 2.3.4 mapreduce 处理框架

2.4 标准化处理

由于各指标间的量纲相差较大，影响后续建模效果，对保留下来的变量数据进行标准化处理。

三、模型假设

为了便于问题的研究，对题目中某些条件进行简化及合理假设。

（1） 由于处理的文本数据比较庞大，假设文本主题与文本详情的内容都是相关的，对研究问题不会造成影响。

（2） 因为收集到的答复和评价都是经过了一定的时间针对相关问题进行的，所以假定其相关性和可解释行都是有一定程度的关联。

（3） 简化模型，对答复意见对应的问题做一个高频词筛选，转化词向量，模拟向量相似拟合度对相关问题答复的一个聚合，从而提高答复完整性。

四、任务一

4.1 基于自然语言处理对文本内容进行清洗

4.1.1 数据预处理

本文通过对附件 2 的留言详情进行过滤和查重，去除掉与“智慧政务”留言详情主题无关的冗余信息形成原始语料，对原始语料内容进行清洗，去掉各种与分析内容无关的信息，解决文本语义带来的词语交叉[1]。原始的留言内容经过数据预处理后，格式更为统一规范，为后续的分词奠定了良好的基础。图 1 为文本分类的基本流程。

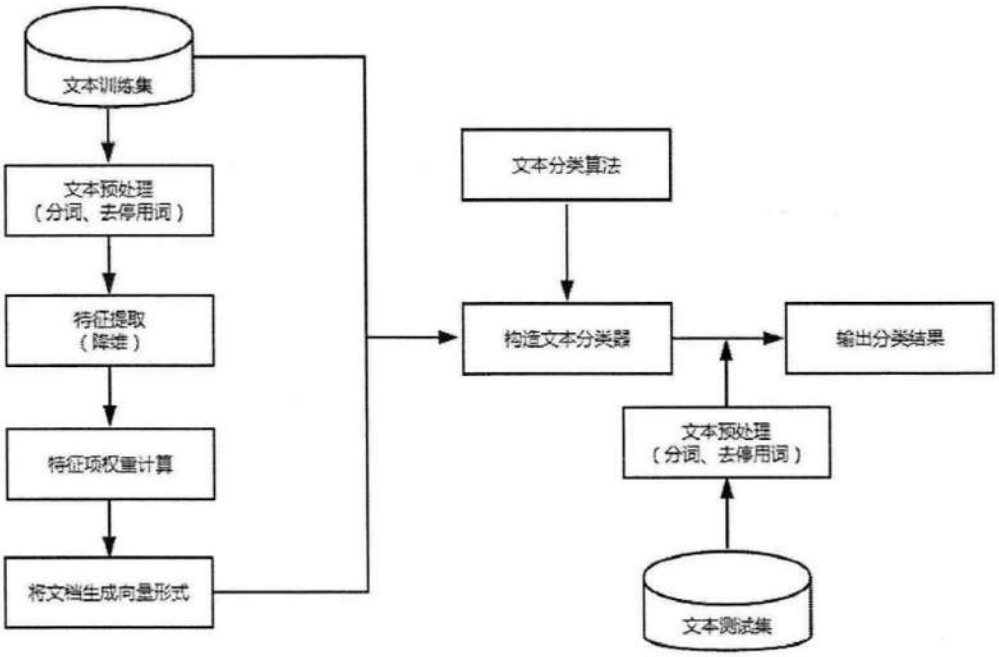

图 1 文本分类流程图

4.1.2 文本特征选择

文本分类本质上都是以数据特征提和最优特征匹配为核心，一般包括数据预处理、文本表示、特征选择、构造并训练分类器、类别预测这 5 步[2]。因为中文不能像英文一样按照空格进行分词，所以在预处理阶段需要使用专门的中文分词算法对文本进行分词处理，并且需要根据停用词表来剔除分词结果中的

停用词。在文本表示方面，词袋模型（bag-of-words）是在文本分类中被广泛应用的表示方法，这种方法不考虑文法和词的顺序，只包含最基础的词频信息。对于特征选择，比较常用的方法有卡方统计量、信息增益、互信息量、TF-IDF等。构造分类器时，常用的传统机器学习分类方法有支持向量机（SVM)、朴素贝叶斯分类法（NBC)、K-最近邻法（KNN)、决策树法（DT）等。上述方法都是以词袋模型为基础的传统的机器学习方法，有很多难以根除的固有弊端。本文将使用 word2vec 技术生成的词向量来代替传统的词袋模型并借助信息提取技术和卷积神经网络来建立关于留言内容的一级标签分类模型，以期获得更优越的性能[2]。

4.1.3 中文分词处理

在自然语言处理过程中，为了能更好地处理句子，往往需要把句子拆开分成一个一个的词语，这样能更好的分析句子的特性，这个过程叫就叫做分词。由于中文句子不像英文那样天然自带分隔，并且存在各种各样的词组，从而使中文分词具有一定的难度。不过，中文分词并不追求完美，而是通过关键字识别技术，抽取句子中最关键的部分，从而达到理解句子的目的。

由于中文不像英文以空格作为单词之间的分隔符，中文词汇之间没有明确的界限，因此需要先对留言内容进行中文分词处理，以词作为留言内容的组成要素。在本文中，我们采用 Python 第三方库中的 jieba 分词器里的精准分词模式作为分词工具以及停用词处理原则。其处理过程可以用一下流程图 2 表示。

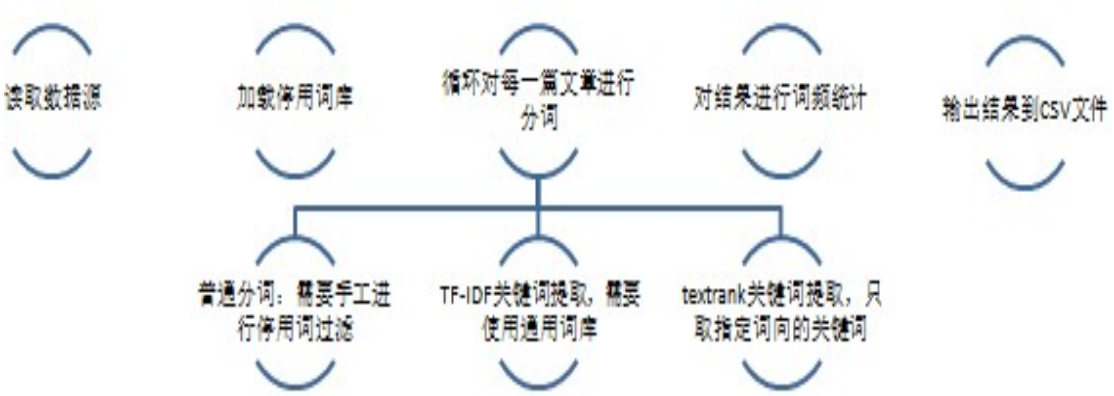

图 2 分词处理的过程

以留言详情“A3 区大道西行便道，未管所路口至加油站路段，人行道包括路灯杆，被圈西湖建筑集团燕子山安置房项目施工围墙内。每天尤其上下班期

间这条路上人流车流极多，安全隐患非常大……”首先去除留言详情中所有的标点符号及无关内容，则信息变化为“A3 区大道西行便道，未管所路口至加油站路段，人行道包括路灯杆，被圈西湖建筑集团燕子山安置房项目施工围墙内。每天尤其上下班期间这条路上人流车流极多，安全隐患非常大”；然后进行分词处理，结果为“A3 区 大道 西行 便道 未管 所 路口 至 加油站 路段 人行道 包括 路灯杆 被 圈西湖 建筑集团 燕子山 安置 房项目 施工 围墙 内 每天 尤其 上下班 期间 这条路 上人流 车流 极多 安全隐患 非常 大”。分词完毕之后，我们还要去除留言完本中的停用词，减少冗余，使文本分类更加准确。插件的停用词有“的”、“如果”、“至”、“非常”、“极多”、“期间”、“被”等对留言内容识别不重要的词。上述留言在去掉停用词后最终的分词结果为“A3 区大道 西行 便道 未管 所 路口 加油站 路段 人行道 包括 路灯杆 每天 上下班 这条路 安全隐患 大”，取得了较好的分词效果。

4.2 使用 Word2vec 技术将数据向量化

词的向量化是指将语言中的词转化成便于计算机处理的数字化表示。Bengio 提出的 NNLM 与 Hinton 等提出 Log-Linear 模型都是使用神经网络来获取词向量的杰出代表[2]。而广为人知的 Word2vec 模型就是借鉴于这两者，是一种更为简洁高效的词向量模型。Word2vec 技术是深度学习技术在自言语言处理应用上个一个关键突破。

4.2.1 word2vec 模型

word2vec 模型本质上是一种简单的神经网络。当网络训练完成后，输入层和隐藏层之间的权重矩阵，就是我们所需的词向量映射表[2]。一般分为 CBOW 与 Skip-Gram 两种模型。

经过查阅相关的资料，我们可以了解到文本信息需要被编码成数字信息才能进行计算处理。然而传统的模型使用基于 one-hot 编码的方法的 BOW（bagofwords）模型，该方法通过构建词典，统计文本的词频信息，对文本进行编码[3]。但是，这种模型存在一定的缺点，one-hot 模型的编码方法孤立了每个词，无法表达出词之间的关系，导致语义信息的丢失。而且，当词的种类过多时，还会带来维度爆炸的问题。因此由于我们处理的数据比较庞大，词的种类过多，可能会带来维度爆炸的问题。所以在这个时候我们可通过将结果 one-hot 编码的词，映射到一个低维的空间，并确保语义信息没有发生变化。也就是我们目前的主流词分布模型 word2vec 模型。

word2vec 包含两种模型，分别是 CBOW 与 Skip-Gram。CBOW 模型通过输入中心词相关的词的词向量，输出中心词的词向量。Skip-Gram 则相反，通过输入中心词的词向量，输出上下文的词向量[3]。如图 2 所示。两种模型 的图形可以用以下两个图形表示，如图 3。

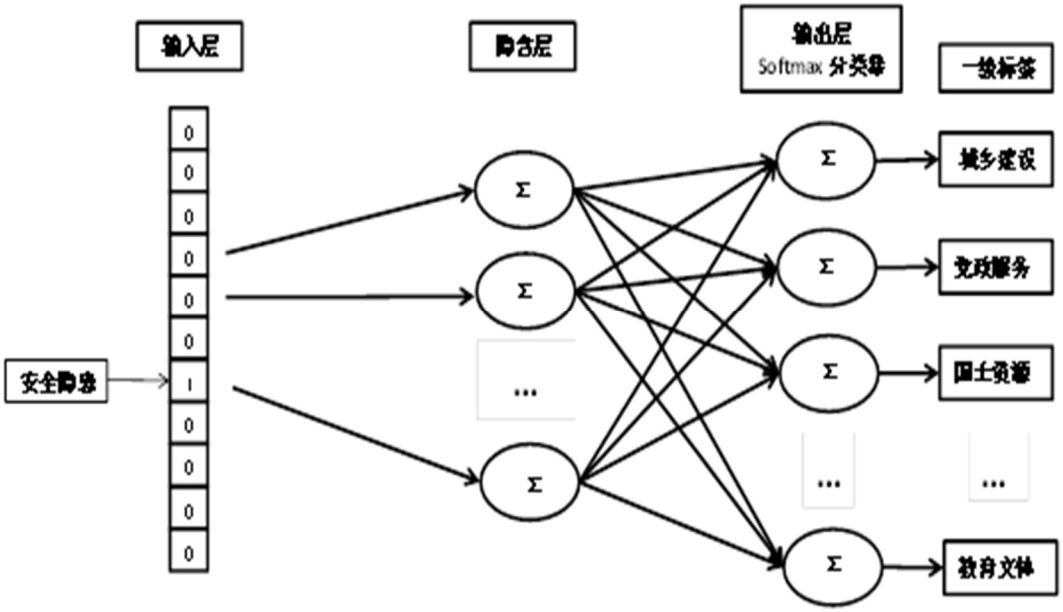

图 3 Skip-Gram 网络结构图

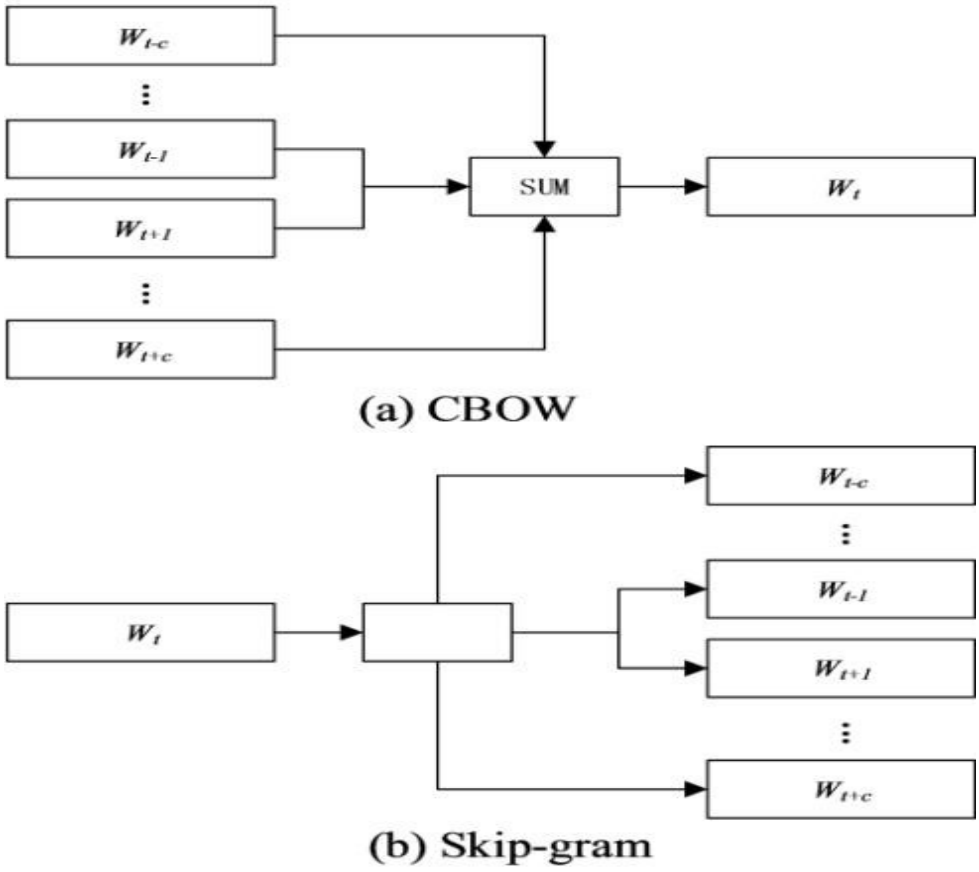

图 4 word2vec 两种模型

其中其中 CBOW 模型利用词 wt 的前后各 c 个词来预测当前词，如图 4(a)所示；Skip-gram 模型则是利用 wt 预测其前后各 c 个词，如图 4(b)所示。

4.2.2 文本词向量化

为了更好地实现对评论文本进行聚类，本模型将主题信息融合到评论文本词向量训练的过程。在从文本中提取文本摘要之后，根据单词向量模型将语句转换为向量。中文维基百科语料库和 word2vec 工具用于训练模型。对于语句 $X = W_1 + W_2 \ldots W_n$，对于其中⊕任意一个词项w_1=（v_1，$v_2 \ldots v_d$），其中的代表词的维度。语句 X 的向量化表示为 S=（$W_1 \oplus W_2 \oplus \ldots \oplus W_n$）。其中⊕是连接运算符。

因此，语句 X 被转换为一串由词向量顺序排列而成的向量。同样，对于每个文本$A = （X_1 \oplus X_2 \oplus \ldots \oplus X_k$），其中 K 表示语句 X 的重要程度排序，也就是说X_i为文本第 i 重要的语句。将文本转化为向量形式后，就可以用向量化的数据来进行神经网络的训练[2]。

4.3 卷积神经网络模型研究

卷积神经网络是由卷积层，池化层和全连接层组成。卷积层通过卷积计算来提取数据的特征。池化层则从卷积层提供的特征中选取最优特征，之后输出给全连接层进行处理。图 1 是本研究所采用的卷积神经网络的结构[2]。

CNN 与其他神经网络有所不同，主要是 CNN 采取了局部连接和权值共享技术，对局部微小的特征更加敏感，更有利于提取留言内容的文本信息特征。通过对留言文本进行卷积和池化操作，可以在词和词的位置信息之间提取出更多的抽象特征值和相关的语义信息。我们将留言内容的 CNN 模型建立如图所示，一词向量作为输入，分类标签做输出，训练 rnn(循环神经网络)神经网络得出分类模型。 如图 5 所示。

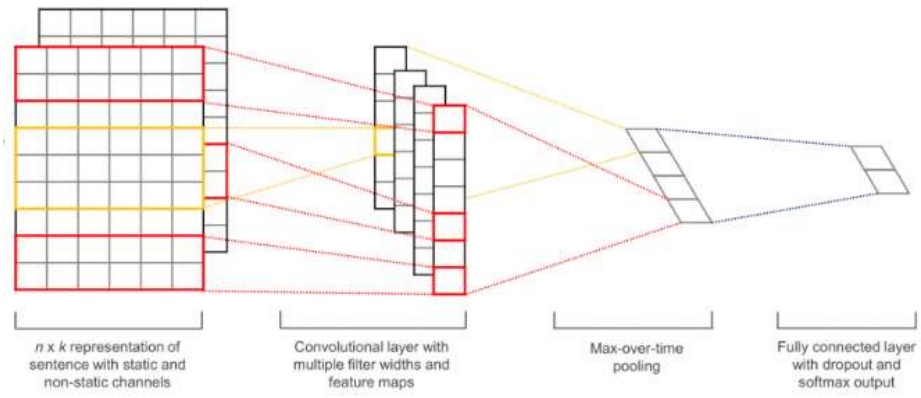

图 5 神经网络模型

4.3.1 输入层设计

在利用卷积神经网络进行训练过程中，因为使用梯度下降方法来进行学习，卷积神经网络的输入特征需要在输入层进行标准化处理。因此我们在处理过程中将文本中经过分词处理以后的词对应的词向量依次排列形成特征矩阵作为输入数据传入卷积神经网络进行训练[4]。每个词向量存储在利用 SkipGram 网络结构提前训练好的词向量模型中，假设文本中有 n 个词，每个词向量维度为 v，那么这个特征矩阵就是 n*v 的二维矩阵[1]。

假设一条留言文本结果分词处理后由“A3 区大道　西行便道 未管所 路口 加油站 路段　人行道　包括　路灯杆 每天 上下班 这条路　安全隐患　大”。中这 13 个词语的词向量依次为Ω_1，Ω_2，… Ω_{13},按照词组词组的顺序作纵向排列，就得到一个表示该留言的特征矩阵 Ω，可以表示为：

$$\Omega = \begin{pmatrix} \Omega_1 \\ \Omega_2 \\ \vdots \\ \Omega_{13} \end{pmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} & \cdots \omega_{1k} \\ \omega_{21} & \omega_{22} & \cdots \omega_{2k} \\ \vdots & \vdots & \vdots \\ \omega_{131} & \omega_{132} & \cdots \omega_{13k} \end{bmatrix} \qquad (1)$$

(1)式的排列方式可以通过图像形象化地表示为图 2 的文本输入层所示的格式。

然而为了便于后续的卷积层和池化层提取出更加抽象的高层次的文本特征，我们需要将各条留言的特征矩阵设置为同一大小。我们取特征矩阵的宽度为各个词组的词向量维数 k。因为留言短信内容的长短不一，特征矩阵的高度应该由留言数据集 A 中的各条留言在经过中文分词处理后，词组数目最多的一条短信含有 m 个词，则该留言可由 m 个 k 维向量按照词组的顺序进行纵向排列为 m×k 的特征矩阵表示。

所以，m 只是所有留言数据 D 中含有含有词汇量最多的一条留言内容的尺度，当留言内容所含有的词汇数目小于 m 的短信，我们则把他作零处理。也就是当含有 b（b<m）个词的任意一条留言与 y,它的特征矩阵中的 b 行由这 b 个词的词向量表示，后面的 b-m 行则用 0 表示。因此，我们可以将任意一条留言内容 b 补 0 后后的特征矩阵 Ω 可以表示成[4]：

$$\Omega = \begin{pmatrix} \Omega_1 \\ \Omega_2 \\ \mathrm{M} \\ \Omega_b \\ \Omega_{b+1} \\ \mathrm{M} \\ \Omega_m \end{pmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} & \mathrm{L} & \omega_{1k} \\ \omega_{21} & \omega_{22} & \mathrm{L} & \omega_{2k} \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} \\ \omega_{b1} & \omega_{b2} & \mathrm{L} & \omega_{bk} \\ 0 & 0 & \mathrm{L} & 0 \\ \mathrm{M} & \mathrm{M} & \mathrm{M} & \mathrm{M} \\ 0 & 0 & \mathrm{L} & 0 \end{bmatrix}$$

4.3.2 卷积层设计

通过内部包含的卷积核进行特征提取，特征提取的计算方法为[1]:

$$S_i = f(C_{h*v} * T_{i:i+h-1} + b)$$

其中C_{h*v}代表卷积核，行数 h 代表卷积核窗口大小，列数 v 为词向量维度，T 为文本特征矩阵，每个卷积核会依次与 h 行 v 列的特征矩阵做卷积操作，b 为偏置量[1]。f 为神经元激活函数，在训练过程中为了防止神经元特征信息丢失以及克服梯度消失问题，设计中采用 LeakyReLU 方法[1]作为激活函数:

f（x）=max（0，x）+ Ymin（0，x）为固定较小常数

我们通过卷积核特征提取后得到的特征图可以表示为：

$$S=[S_1, S_2, \ldots, S_{m-h+1}]$$

因为在不同的留言内容中，其前后的语义关联可能不一样，然而为了使从留言文本中提取得到较为完善的特征。基于这个目的，我们对文本处理采用了三种窗口高度不一样的卷积核提取相应的局部语义特征[4]。通过观察文本数据可以发现，每一条留言的内容的尺度一般在十几个字到几十个字之间，因此我们将卷积核的高度分别设置为 3、4、5，最终得到了 3 种不同粒度的特征。CNN 提取特征的具体流程如图 6 所示。

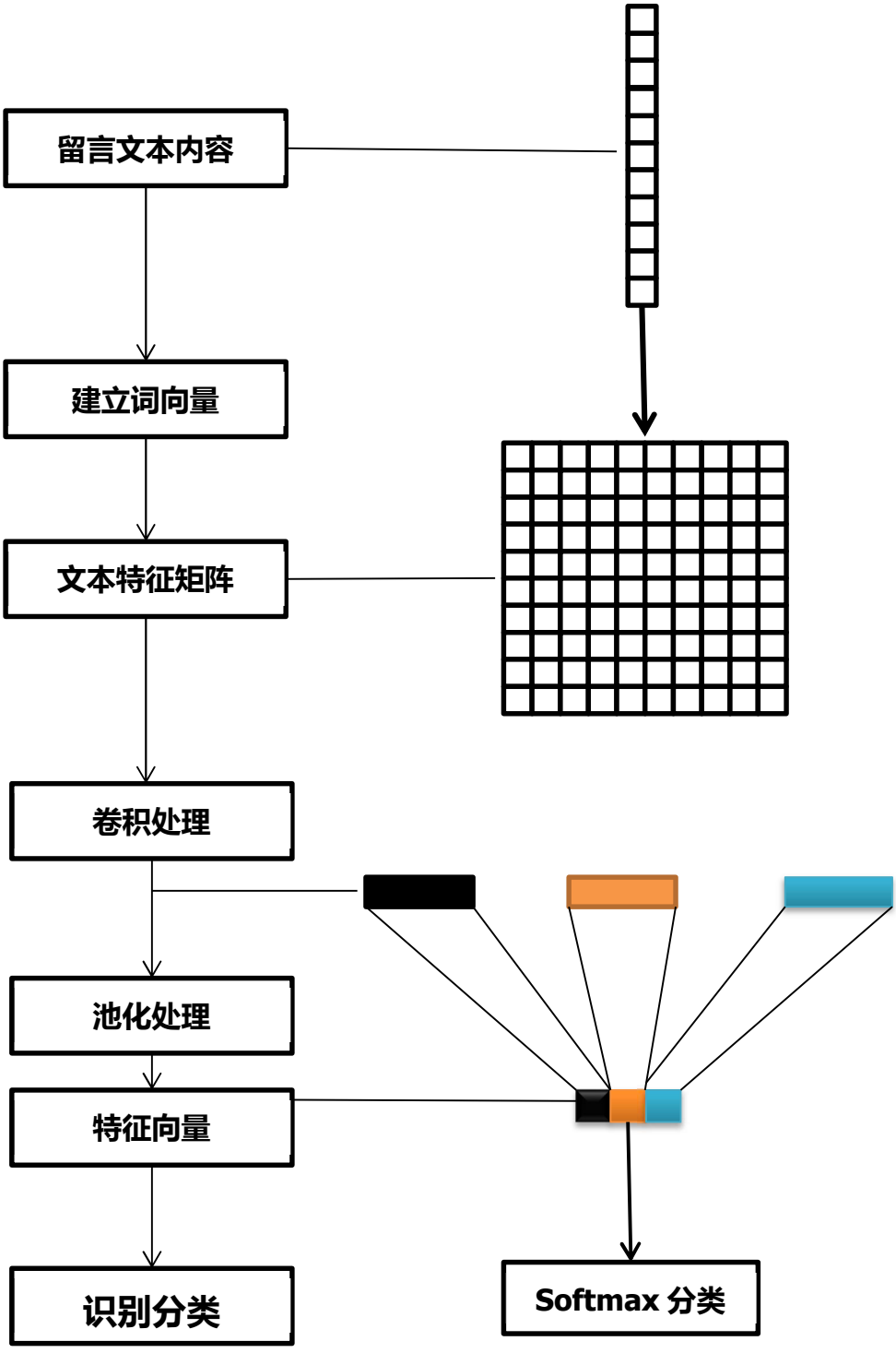

图 6 CNN 提取 特征流程图

由于在卷积层的设计过程中，我们考虑到一个卷积核在提取特征时存在不充分性的问题，因此将三种卷积核的大小分别设置为 3×k 、4×k 、5×k。经过卷积核运算之后，得到 3 种粒度大小的特征图，分别是（m-2）×1、（m-3）×1 和（m-4）×1。其处理过程为图三所示[4]。每个卷积核的操作模式设置为相同，及设置结果每种特征图各提取出张。最终在卷积层的输出端得到共张特征

图。每组尺寸卷积核各 270，即对每一条留言内容的文本特征作为特征矩阵输入，一种卷积核可以得到 270 个特征图，卷积层输出了 589 个特征图。

4.3.3 池化层设计

因为在卷积层进行特征提取后，特征图的维度还是很高，这时我们可以将特征图传递至池化层通过池化函数进行特征选择和信息过滤。通过池化函数将特征图中单个点的结果替换为其相邻区域的特征图统计量，池化过程与卷积层扫描特征图的过程相同[1]。工作流程如图 7 所示。为了保证准确性，我们在实验中采用最大池化函数(MaxPooling)对卷积核获取的特征保留最大值同时放弃其它特征值，即我们对每一个特征图只选取一个最大值 Z，Z 的表达式为

$$Z=\max\{Z_1, Z_2, \ldots, Z_{m-h+1}\}$$

其中 Z 为留言内容的最优居部特征。

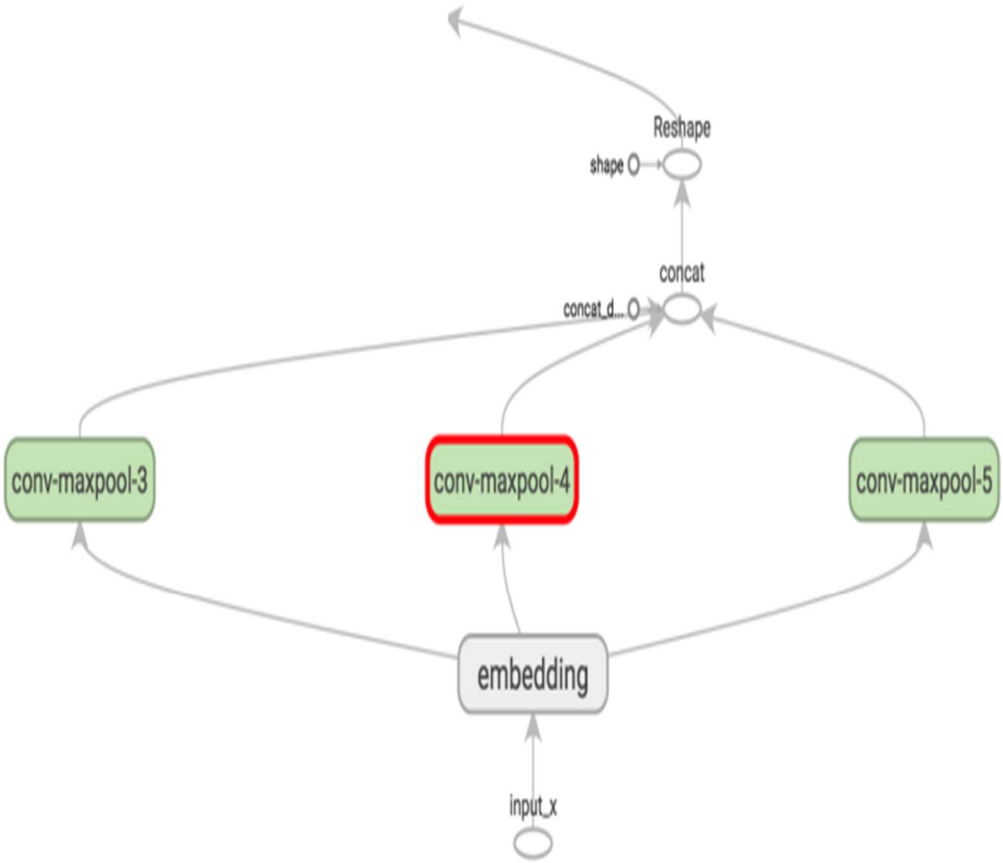

图 7 池化层工作流程

4.3.4 全连接层设计

所谓的全连接层设计就是对提取的特征进行非线性组合得到输出，全连接层本身不具有特征提取能力，主要用来整合池化层中具有类别区分性的特征信息，在实验中采用 LeakyReLU 函数作为全连接层神经元的激励函数[1]，如图 8 所示。

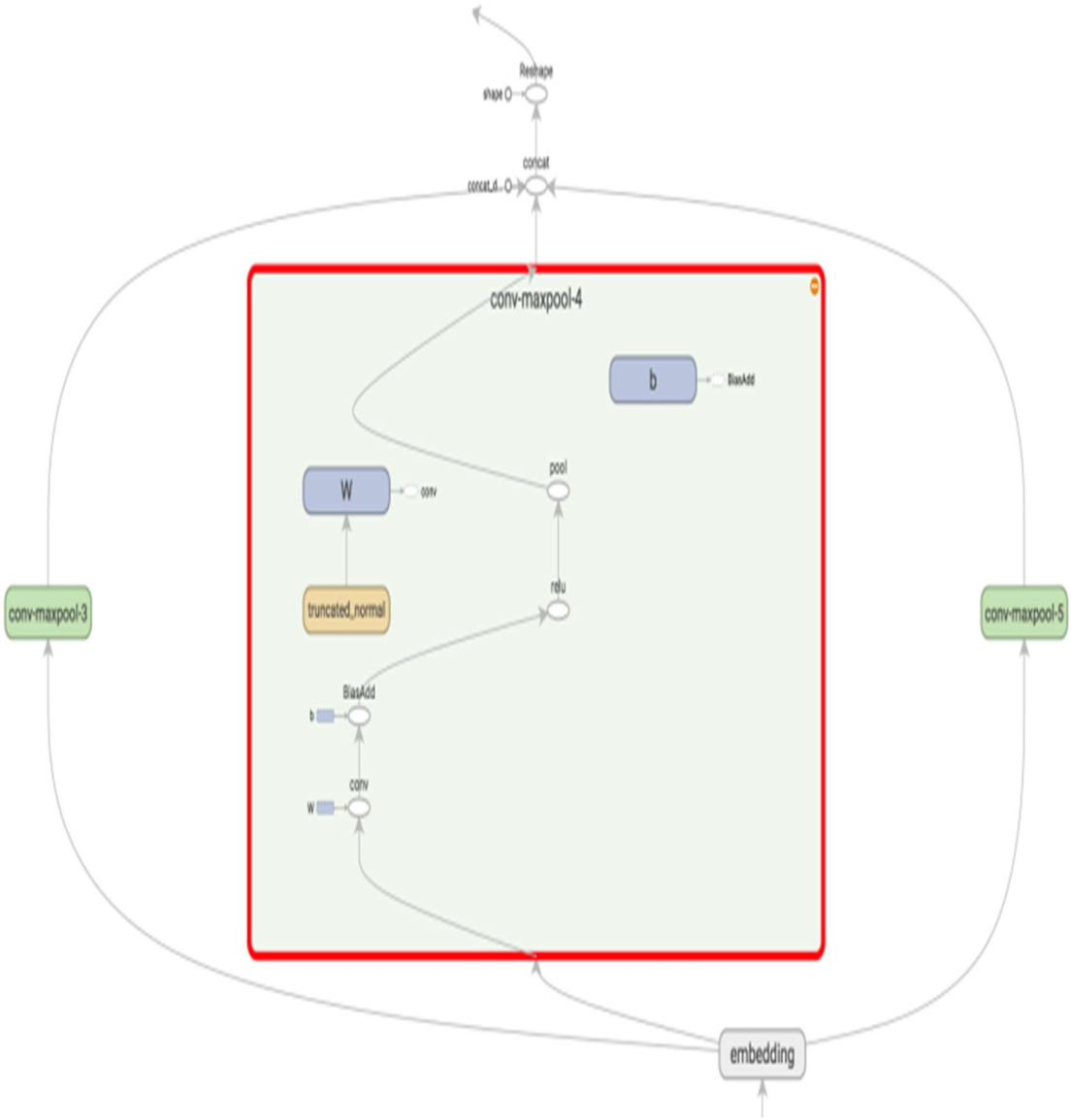

图 8 全连接层设计

4.3.5 输出层设计

输出层设计是指可以使用多类交叉熵函数(Multiclass Cross Entropy)作为损失函数以及归一化指数函数(Softmax)[1]作为激活函数输出分类标签，以词向量作为输出标签，完成建立一级标签分类模型。其工作流程如图 9 所示。

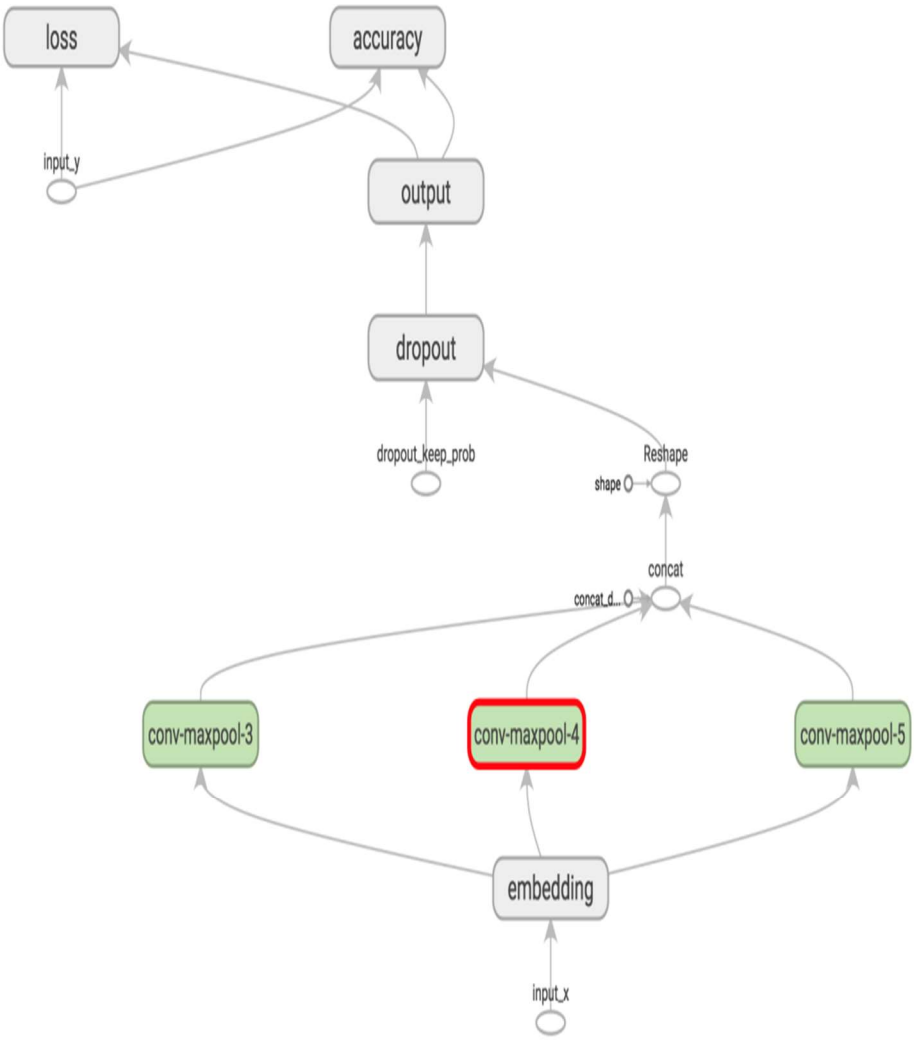

图 9　输出层设计

4.4 实验分析

4.4.1 实验环境

实验工具	型号或版本
CPU	锐龙 3500
处理器	Intel（R） Celeron（R）
操作系统	Win10
编程语言	Python 3.7
深度学习框架	Tensorflow 1.0
中文分词	jieba 0.9

4.4.2 实验数据

本文训练词向量所用的维基中文语料包含 9211 个中文文本。所有文本经过词向量训练之后，一共有 10 万多个中文词组。

参与实验的留言短信文本数据从总体上观察，有一些语句冗杂，与留言内容并无直接关系，有一些留言可以根据留言的三级分类标签做出明显的判断，即具有特征标签。

4.4.3 实验设计

本文将词向量的维度设置为 100，即 k=100。在训练词向量的时候为了避免关联到更多语义不相关的词汇以及缩短训练的时间，我们可以将窗口的大小设置为 5，即 c=5。在实验开始之初，我们将群众的留言详情通过中文分词、去除停止词以及标点符号等操作之后，长度最长的一条留言内容含有 100 个词，即 m=100；因此，我们建立每条留言内容都必须表示为 100×100 的特征矩阵。

我们通过借鉴其他的论文文献，在以词向量组成表示留言文本的特征矩阵的基础上，同时结合了训练训练神经网络的模型建立关于留言内容的一级标签分类模型。

具体的实施方案如下：

(1)词向量+CNN 模型：将表示短信内容的文本的 100×100 特征矩阵作为 CNN 的输入参与到模型的循环测试过程当中；

(2)词向量+传统的机器学习模型：对于相同的数据，我们采用同样的分布式特征提取方法获取每一个文本内容的词向量，其中参考我们的留言分级标签体系的三级标签，将文本的关键词作为特征向量。我们查找相关的文献，采用相关的分类模型包括朴素贝叶斯、逻辑回归、支持向量机和随机森林。朴素贝叶斯是基于贝叶斯定理与特征条件独立假设的分类方法[1]。

4.4.4 模型的训练测试

本文采用了 mini-batch 梯度下降的方法进行 CNN 模型的训练，也就是意味着我们将一部分的留言内容参与迭代的过程。以为我们设置的特征矩阵为 100×100，及留言容量为 100000。为了减少失误以及保证实验结果的准确性，我们将每批样本的大小设置为 1000。

同时在模型的测试过程中，我们使用了多类交叉熵函数(Multiclass Cross Entropy)交叉函数来衡量模型，减少拟合程度。

假设留言内容为（Ω，Y），其中 Ω 表示文本内容的特征矩阵，Y 表示一级标签，因为一级标签一共有 15 钟类型，包括了城乡建设、党务政务、国土资源、环境保护、纪检监察、交通运输、经济管理、科技与信息产业、民政、农村农业、国贸旅游、卫生计生、政法、教育文体、劳动和社会保障。当留言内容有这些词向量时，我们将其表示为 1，其他为 0，因此，Y 表示类别标签可以表示为：

$$
Y=\begin{cases}
(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0) & \text{城乡建设}\\
(0,1,0,0,0,0,0,0,0,0,0,0,0,0,0) & \text{党务政务}\\
(0,0,1,0,0,0,0,0,0,0,0,0,0,0,0) & \text{国土资源}\\
(0,0,0,1,0,0,0,0,0,0,0,0,0,0,0) & \text{环境保护}\\
(0,0,0,0,1,0,0,0,0,0,0,0,0,0,0) & \text{纪检监察}\\
(0,0,0,0,0,1,0,0,0,0,0,0,0,0,0) & \text{交通运输}\\
(0,0,0,0,0,0,1,0,0,0,0,0,0,0,0) & \text{经济管理}\\
(0,0,0,0,0,0,0,1,0,0,0,0,0,0,0) & \text{科技与信息产业}\\
(0,0,0,0,0,0,0,0,1,0,0,0,0,0,0) & \text{民政}\\
(0,0,0,0,0,0,0,0,0,1,0,0,0,0,0) & \text{农村农业}\\
(0,0,0,0,0,0,0,0,0,0,1,0,0,0,0) & \text{国贸旅游}\\
(0,0,0,0,0,0,0,0,0,0,0,1,0,0,0) & \text{卫生计生}\\
(0,0,0,0,0,0,0,0,0,0,0,0,1,0,0) & \text{政法}\\
(0,0,0,0,0,0,0,0,0,0,0,0,0,1,0) & \text{教育文体}\\
(0,0,0,0,0,0,0,0,0,0,0,0,0,0,1) & \text{劳动和社会保障}
\end{cases}
$$

其中交叉熵函数失约束表示如下所示：

$$L(Y,G(\Omega)) = -[(1-G(\Omega))\ln(1-G(\Omega)) + Y\ln G(\Omega)] + \frac{\gamma}{2}\|\theta\|_2$$

式子中 G 表示我们建立的 CNN 模型，G（Ω）表示模型的输向量，γ为正则项系数，θ为参数向量。

在训练全连接层参数时，为了避免发生过拟合，采取 dropout 策略使部分神经节点失效，即一些已经训练过的参数在每一次更新的时候将被随机选择丢弃[5]。在训练神经网络模型的时候，将 dropout 的值设置为 0.5，选择随机性的丢弃一半的参数[4]。本文以 AdamOptimizer 优化器，参数设置 CNN 模型。如图 10 所示。

参数名称	参数值
卷积核的个数	270
卷积核窗口高度	3,4,5
词向量维度	100
Mini-batch	1000
Dropout	0.5
训练次数	60
学习率	0.4　0.7　　0.9

图 10 CNN 算法参数设置

4.5 评价标准

为了验证 CNN 模型在建立留言内容一级标签分类模型的可靠性，本文采用了 F-Score 对分类方法进行评价：

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

其中P_i为第 i 类的查准率，R_i为第 i 类的查全率。

F-score 是一种衡量特征在两类之间分辨能力的方法，通过此方法可以实现有效的特征选择，但是地质问题往往是多类问题[6]。当 F-score 值越大，表明此特征的辨别力越强。

由于 F-score 通过 SVM 分类模型的分类准确率作为评估准则，能够实现对特征子集的有效选取，提高模型分类器性能 [6]。将数据集中每个特征根据计算后的 F-score 值降序排列，从每个特征根据计算后的 F-score 值降序排列，从 F-score 值高到低依次选取一个特征加入到被选的特征集合中；再应用 SVM 算法对当前选取的特征子集减小评价[2]。其流程图如图 11 所示。

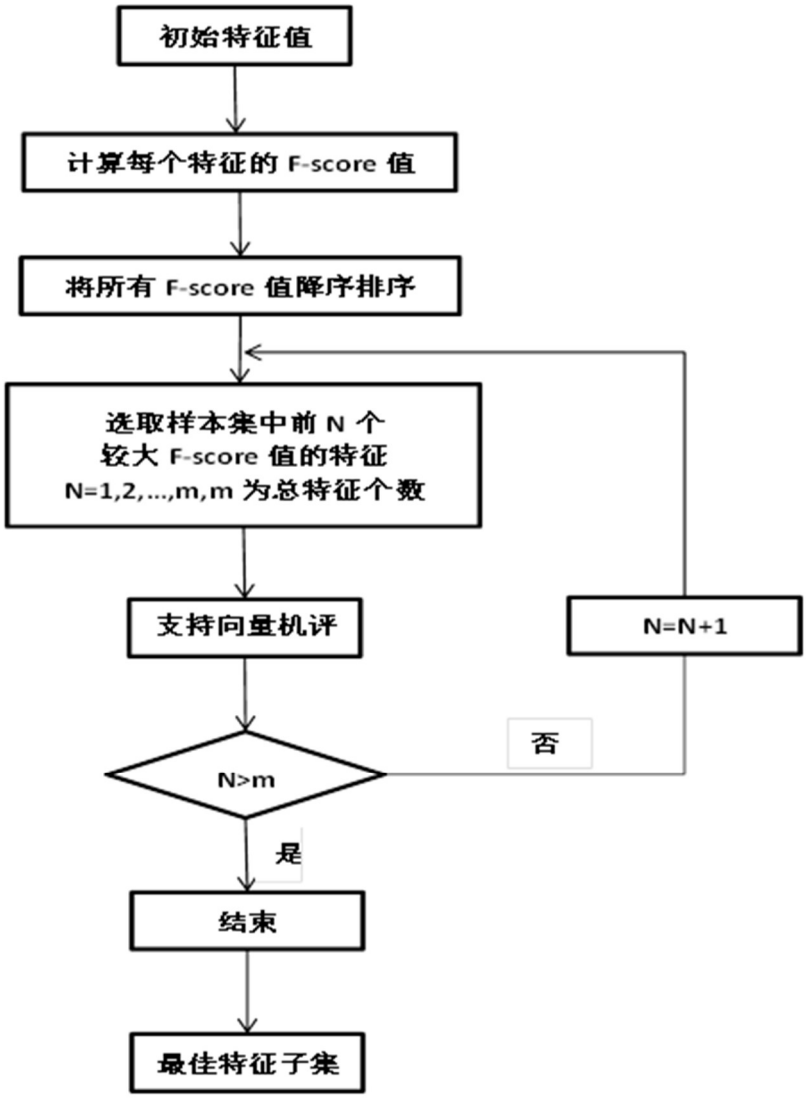

图 11 图特征选择模型

4.6 实验结果分析

通过 CNN 神经网络训练模型，我们可以发现 CNN 的收敛速度与学习率有一定的关系。当学习率太小的时候，此时无法找到好的下降方向，导致训练时间比较长；相反当学习率过大时，则会造成神经网络出现超调或剧烈震动。这时，我们需要设置合理的学习率，以获得较准确的结果[2]。在不同学习率下的准确率变化如图 12 所示：

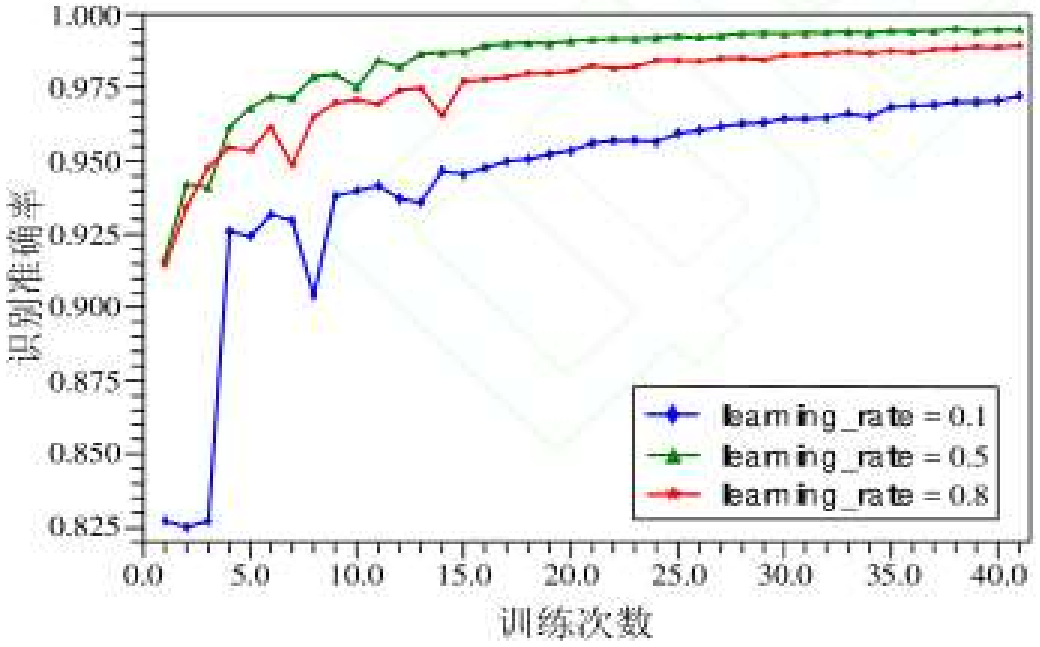

图 12 不同学习率下的准确率变化

从图 11 我们可以看出，当学习率为 0.1 和 0.8 的时候，CNN 神经网络的收敛比较稳定，但是速度比较慢。而学习率为 0.5 的时候，不仅收敛比较快，而且准确率相对比较高。

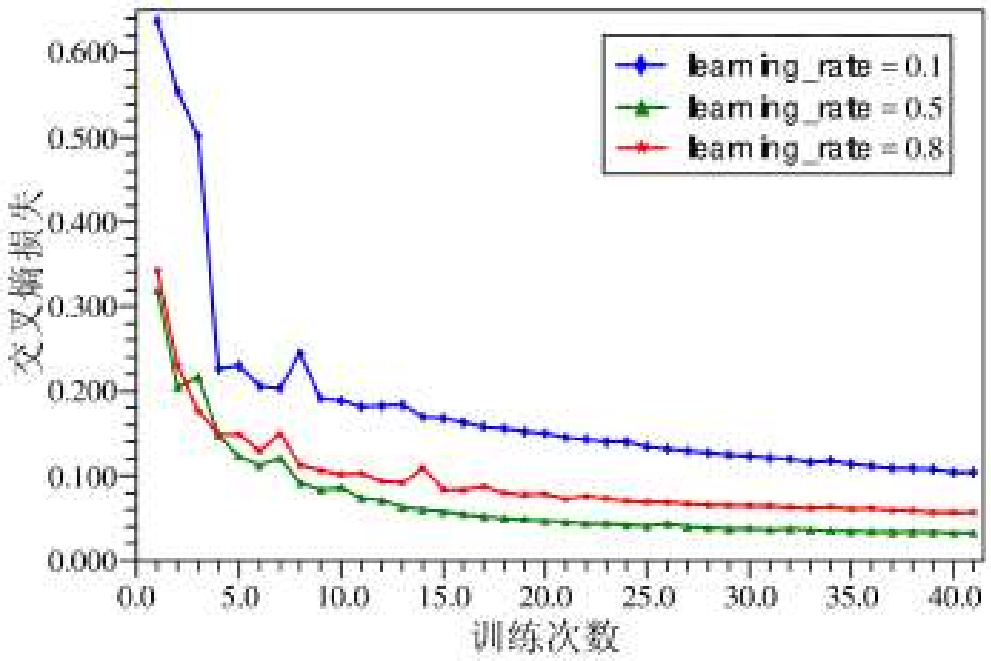

图 12 不同学习率下的损失变化

图 12 表明了随着学习率的不断增大，损失在不断减小；当学习率为 0.05 时，损失收敛最快。

4.6.1 基于 F-score 分类准确率

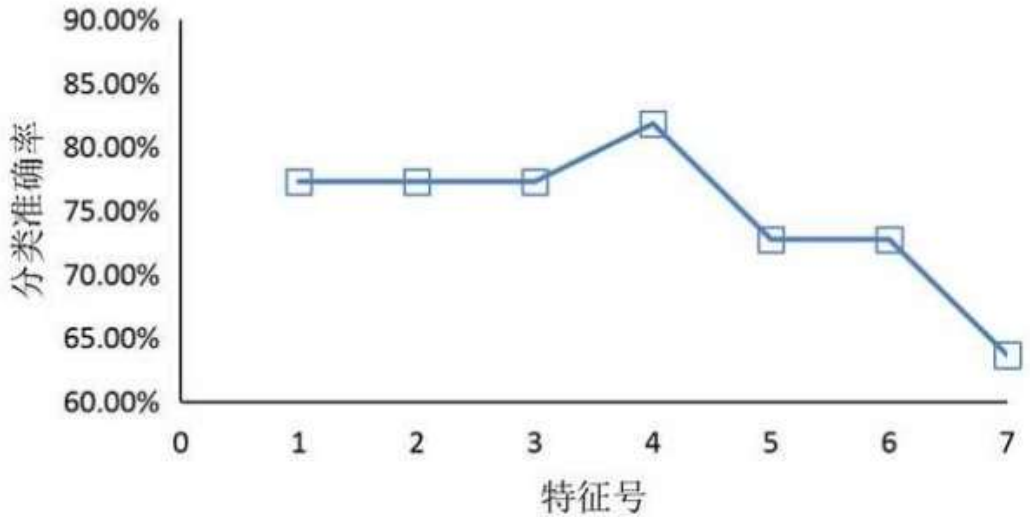

图 13 基于 F-score 分类结果

五、任务二

在某一时间段内群众集中反映的某一问题称为热点问题。比如“A 市 A5 区魅力之城小区业主多次反映了小区临街餐饮店油烟噪音扰民”。为了使相关部门能够及时的发现并有针对性地处理群众留言问题，提升服务效率，我们通过建立 LAD 模型对某一时间段内反映的特定地点或特定人群问题的留言进行归类，同时通过量化评价指标定义合理的热度评价指标，并给出合理的评价结果。

这道题的基本流程如图 14 所示，主要包括数据采集、数据预处理、文本建模、计算相似度、基于 K-means 算法进行文本聚类、聚类得到相应的留言热点问题。其中，文本建模过程包括：Biterm__VSM 特征词建模、LDA 的主题建模。计算相似度包括：计算基于词对特征值的文本相似度、计算基于主题的文本相似度。

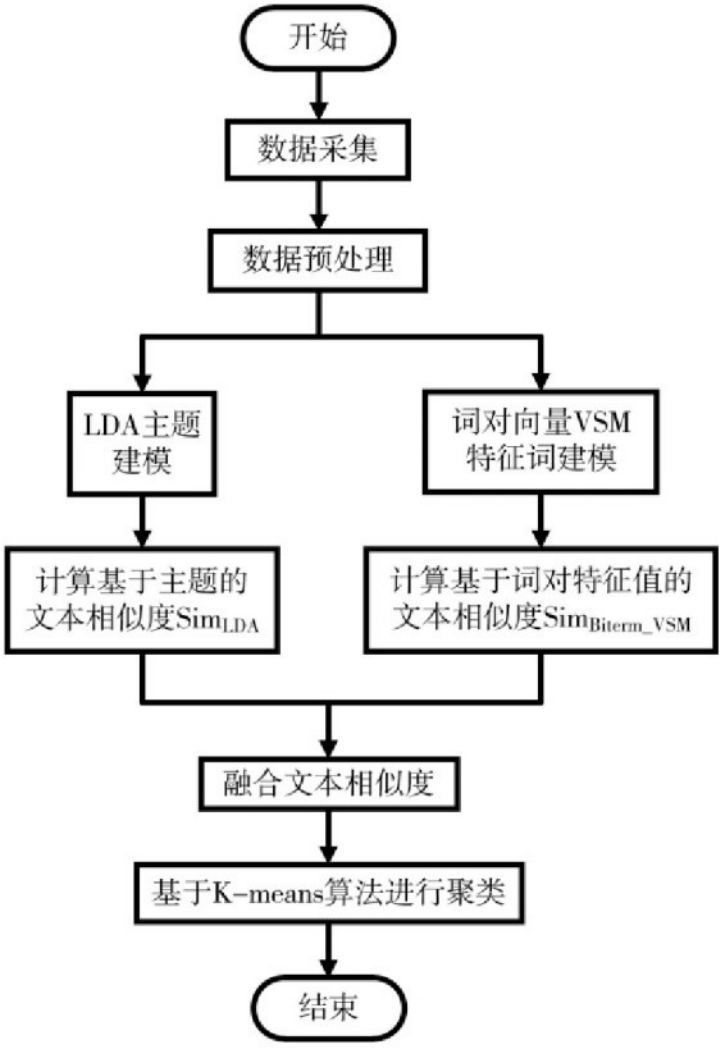

图 14 特征融合的文本聚类分析流程图

5.1 数据预处理

为了控制主题分类质量，减少噪音数据对主题分类的影响，本文分四步对爬取的原始数据进行了预处理[7]。

第一步：数据清洗。将题目提供的原始数据中重复数据、与留言详情无关以及无意义超短文本等无意义数据剔除，最终得到 4327 条留言详情。

第二步：文本分词。使用中科院的分词工具对最终的线上评论数据进行分词。

第三步：去停用词。本研究采用包含 650 个停用词的通用停用词表作为初始停用词表，在进行了多次主题分类实验后，据实验结果将对主题分类无意义的高频词，如尊敬的领导、你好、宵夜、谢谢等增添到初始停用词表，对原始停用词表进行扩展形成了最终的停用词表。

第四步：去除长度小于 2 的词。长度小于 2 的词大都不具有实际意义，因此将这些词从词典中剔除。

5.2 词对向量 VSM 特征词建模

在一句文本中，往往仅用少量的词就可以充分地表示这个句子的语义，因此考虑使用多个词作为一个文本的基本单元，但是同时带来的是高维度，使得模型过于复杂[8]。因此本文使用在一句文本中共现的词对作为文本表示的基本单位。

本文以一句文本中共现的词对为特征使用向量空间模型(Biterm__VSM)将微博文本转化为空间向量，并且使用词频-逆文档频率 (Term Frequency-Inverse Document Frequency,TF-IDF)方法计算每一维的特征权重[8]。在基于词对的向量空间模型中，我们假设文本由不重复的词对组成。因此一个文本可以表示为：

$$T=\{（（C_1^{\,1}，C_1^{\,2}），W_1），\cdots，（（C_m^{\,1}，C_m^{\,2}），W_m）\}$$

其中（$C_1^{\,1}$，$C_1^{\,2}$）为词对，m 为这个文本中词对的数目，W_1为某个词对的权重。本文中的共现词对是由来自同一个文本的两个不同的词组成，不分两者的先后顺序，即($C_i^{\,1}$，$C_i^{\,2}$)=($C_i^{\,2}$，$C_i^{\,1}$)。为了简化模型，方便操作，我们定义一句文本的概念为一句留言。例如：A7 县东四路恒大翡翠华庭路段行人过马路尤为不便。

这个文本预处理之后为“A7 县|东四路|恒大翡翠|华庭路段|行人|过马路|尤为|不便”。其产生的词对集合为{(A7 县，东四路)，(A7 县，恒大翡翠)，…，(过马路，不便)}。另一方面由于数据集中的词对比词在文本中出现的频率要低，因此其权重不使用词对的词频-逆文档频率值(V_{tf-idf})来表示，而是由词对中两个词的V_{tf-} 的和来表示：

$$W(C_i^{\,1}，C_i^{\,2})=V_{tf-idf}(C_i^{\,1})+V_{tf-i}\quad(C_i^{\,2})$$

5.3 LDA 主题模型

LDA 模型是一种文档生成模型，其概率图模型如图 15 所示，它将文档表示为主题的概率分布，而主题表示成词的概率分布，因此 LDA 可以被用来进行文本特征提取。LDA 的输入是文本的 one-hot 编码，输出是文档的主题分布、主题的词分布[3]。LDA 模型可以描述如下[3]。

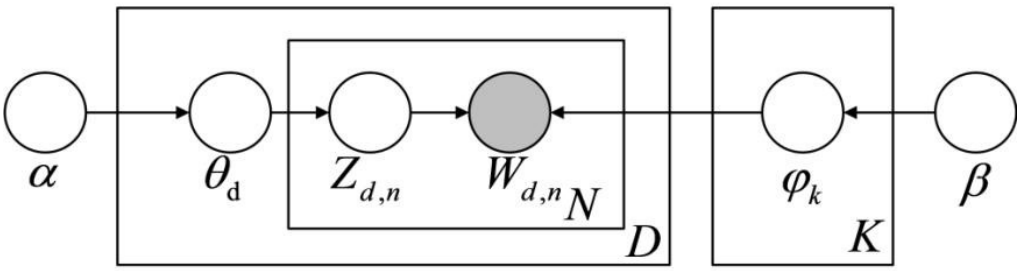

图 15　LDA 概率图模型

1）文档的主题先验分布服从参数为 α 的 Dirichlet 分布，其中文档 d 的主题分布为 θ d=Dirichlet(α)

2）主题中的词的先验分布服从参数 β 的先验分布，其中主题 k 的词分布为 φ k=Dirichlet(β)

3）文档 d 中的第 n 个词，从主题分布获得其主题编号分布为 zdn=multi(θ d)

4）文档 d 中的第 n 个词分布 wdn 的分布为 wdn=multi(φ zdn)

其中 D 是训练数据集的大小，N 是一条训练数据的大小，K 是主题数。

通过模型假设可以发现，已知每个文档的文档主题的 Dirichlet 分布与主题编号的多项式分布满足 Dirichlet-multi 共轭，使用贝叶斯推断的方法得到文档主题的后验分布[3]。同样已知主题词的 Dirichlet 分布与主题编号的多项式分布满足 Dirichlet-multi 共轭，通过贝叶斯推断得到主题词的后验分布。然后通过使用 Gibbs 采样的方法去获得每个文档的主题分布和每个主题的词分布。

5.3.1 留言主题信息提取

统计语料集的词频信息建立字典，对文本进行 BOW 编码，输入到 LDA 模型中，获得每条评论的主题分布$d_t = [Z_1, Z_2, \ldots Z_k]$，其中 z 为每个主题编号的概率，然后找到每个主题的词分布$t_w = [w_1, w_2, \ldots, w_N]$，其中 w 为字典中每个词的分布概率。则每条留言的主要特征词可以表示为如公式（1）所示[3]。

$$D_W = \{Z_1 * W_1 Z_1 * W_2 \ldots Z_2 * W_1 Z_2 * W_2 \ldots \ldots Z_k * W_N\} \quad (1)$$

我们通过设置阈值，选取D_W中超过阈值的词作为评论文本的主要词特征。

5.3.2 LDA 主题词提取

将留言主题语料集经过预处理后输入到 LDA 模型，得到语料库的主题词概率分布和每条留言的主题概率分布。使用公式（1）计算添加到每条评论中的主

要词特征。根据情感分类准确率选择主题词选取阈值为 0.06，对添加到评论中的主题词进行统计，在指定 LDA 主题总数为 30、35、40、45 时，评论中满足阈值条件的主题词如表 1 所示。

11,	A3 区 青青 家园 小区 乐果 果 零食 炒货 公共 通道 摆放 空调 扰民
12,	拆除 聚美龙楚 西地省 商学院 宿舍 旁 安装 变压器 请求
13,	市利保 壹号 公馆 项目 夜间 噪声 扰民
14,	市 地铁 号线 星沙 大道 站 地铁 出入口 设置 不合理
15,	A4 区 北辰 小区 非法 住 改商 何时能 解决
16,	请 K3 县 乡村 医生 发 卫生室 执业 许可证
17,	A7 县 春华 镇 石塘 铺村 党员 家开 麻将馆
18,	咨询 异地 办理 出国 签证
19,	投诉 市 温斯顿 英语 培训 学校 拖延 退费
20,	A6 区乾源 国际 广场 停车场 违章 乱建 现象
21,	A7 县 时代 星城 幢 非法经营 家庭旅馆
22,	A2 区佳兆业 水新 小区 垃圾 无人
23,	市 沙坪 街上 有无 证 理疗 馆 骗取 老人 钱财
24,	市德鸿 餐饮店 拖欠工资 员工 维权 难
25,	市长 房云 时代 小区 三期 要建 垃圾站
26,	市松雅湖 东方 航标 栋 楼有 传销 窝点
27,	希望 市政府 出台 解决 落实 退休 教师 各项 补贴 长效 办法
28,	举报 市仕弘 教育 培训 机构 涉嫌 欺诈
29,	A2 区政府 东门 万 芙 路段 改装车 飙车 真的 扰民
30,	请 依法 解决 A7 县 黄花镇 梁坪村 黄泥 岭 山地 建房
31,	A7 县 橄榄 城 小区 孩子 到泉塘 小学 上学
32,	投诉 滨河 苑 广铁 职工 购房 霸王
33,	市 万家 丽 南路 丽发 新城 居民区 搅拌站 扰民
34,	市星沙 城区 旧城区 棚户 改造 项目
35,	A2 区 先锋 派出所 办个 签证 拒收 现金
36,	A3 区谷园 路 号 维也纳 智好 酒店 卖淫 团伙
37,	咨询 A7 县椰 梨 龙华 安置 区 外围 马路 修复

表 1 LDA 提取主题词实例

对分词后的数据用词云表示如图 16 所示，其中关于“小区”，“扰民”，“街道”等为留言内容的热点词汇。

图 16 数据集词云

5.4 特征融合的文本相似度计算

在本文中，为了确定文本相似度是进行下面文本聚类分析的关键一步。因此我们利用了线性组合的方式将基于 TF-IDF 的词对向量空间模型和基于 LDA 主题模型结合，得到文本相似度，即本文的特征融合文本相似度[8]。

TF-IDF 倾向于过滤掉常见的词语，保留重要的词语[8]。首先使用 TF-IDF 算法，找出两个文本的关键词，再通过计算词频，建立权重空间，最终计算余弦值判断文本相似度[8]。该算法中 IDF 的简单结构并不能有效地反映单词的重要程度，使得该模型无法很好地完成对权值调整的功能。

其中，线性结合的计算公式：

$$\text{Sim}（d_1,d_2）=sim_{Biterm_{vsm}}(d_1,d_2)+sim_{LDA}(d_1,d_2) \qquad （2）$$

其中$sim_{Biter\ \ _{vsm}}(d_1,d_2)$为两个文本间词对向量空间模型的文本相似度，simLDA(d1,d2)为两个文本间主题向量空间模型的文本相似度。图 17 为流程相似度流程[9]。

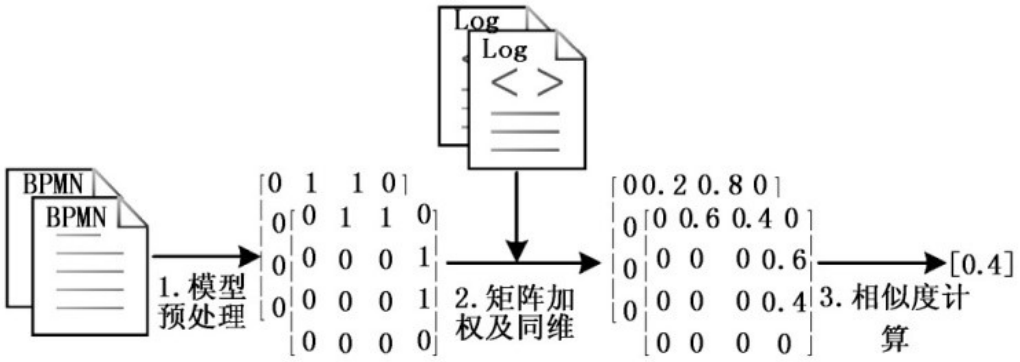

图 17 流程相似度计算流程

5.4.1 词对向量空间模型文本相似度计算

不同的模型需要用不同的相似度计算方法，采用词对的标准化 TF-IDF 值来衡量词对向量空间模型中的文本，采用欧氏距离来计算文本的相似度[8]。

计算公式如下（3）所示：

$$sim_{Biter\ \ _{vsm}}(t_1,t_2)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \qquad (3)$$

其中，t_1和t_2分别为两个文本，x_i为文本t_1的第 i 个特征向量，y_i为文本t_2的第 i 个特征向量。

5.4.2 主题向量空间模型文本相似度计算

采用服从 Dirichlet 分布的主题概率向量来表示 LDA 主题模型中的文本，同样采用欧氏距离来计算文本的相似度，其计算公式如下（4）所示[8]：

$$sim_{LDA}(p,q) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2}$$

其中，p 和 q 为两个文本的主题概率分布，ai 和 bi 为两个文本中主题 θ i 的概率分布。

5.5 特征融合的文本聚类算法

本文聚类算法采用的是经典算法 K-means 算法[8]，该算法思想简单，易于实现，可以快速有效地处理大规模数据。

K-means 聚类算法的基本思想如下：

输入：簇数目 K，特征融合的文本向量矩阵

输出：簇的集合 D

步骤：

(1)从集合 D 中随机选择 K 个数据点作为话题簇的初始聚类中心点；

(2)计算每个微博向量与聚类中心点的距离，并将该微博向量分配到最近的中心点；

(3)重新计算 K 个簇的聚类中心并更新；

(4)重复上面的 2 个过程，直到话题簇的中心点不再变化，或者达到收敛的条件停止算法；

(5)输出聚类簇的结果。

5.6 实验分析

5.6.1 实验环境

本文的实验是利用 PyCharm 平台下的 Python 语言实现的。所有实验均在一台操作系统为 64 位的 Windows 10 家庭中文版的 Lenovo 台式电脑上，处理器为 Inter（R）Celeron（R） CPU N3160 @ 1.60 GHz 1.60 GHz，内存为 8 GB。

5.6.2 热度评价指标

热点留言是一个聚类的过程，而且实验使用的数据是无标签的，因而无法使用类似于分类过程中的评价体系，例如准确率、精确率、召回率以及由精确率和召回率得到的 F1 值。因此本文实验的评价指标采用的是聚类的一种评价指标———调整兰德系数(Adjusted Rand Index,ARI)[8]。

兰德系数(Rand Index,RI),RI 取值范围为[0,1]，值越大意味着聚类结果与真实情况越接近。

$$\mathrm{RI}=\frac{a+b}{C_2^n}$$

其中，C 表示为留言内容，K 表示聚类结果，a 表示在 C 和 K 中都是同类别的元素对数，b 表示在 C 和 K 都是不同类别的元素对数，C_2^n表示数据集中可以组成的对数。

调整兰德系数(ARI)取值范围为[-1,1]，值越大表示聚类结果和实际的数据集越接近。从广义的角度来说，ARI 是衡量两个数据分布的吻合程度。

$$\mathrm{ARI}=\frac{RI-E(RI)}{\max(RI)-E(RI)}$$

调整兰德系数(ARI)取值范围为[-1,1]，值越大表示聚类结果和实际的数据集越接近。从广义的角度来说，ARI 是衡量两个数据分布的吻合程度。

5.7 对比实验

5.7.1 实验 1

对于该问题的对比实验 1 分别由下面的 4 部分组成：

(1)采用基于词特征的传统向量空间模型对留言内容进行表示，并由 K-means 聚类算法进行进行聚类实验；

(2)采用基于词对特征的向量空间模型对留言内容进行表示，并由 K-means 聚类算法进行进行聚类实验；

(3)采用基于主题向量空间模型对留言内容进行表示，并由 K-means 聚类算法进行进行聚类实验；

(4)采用基于词对特征向量空间模型和基于主题向量空间模型融合来对留言内容进行表示，并由 K-means 聚类算法进行聚类实验。

其实验结果对比如图 18 所示

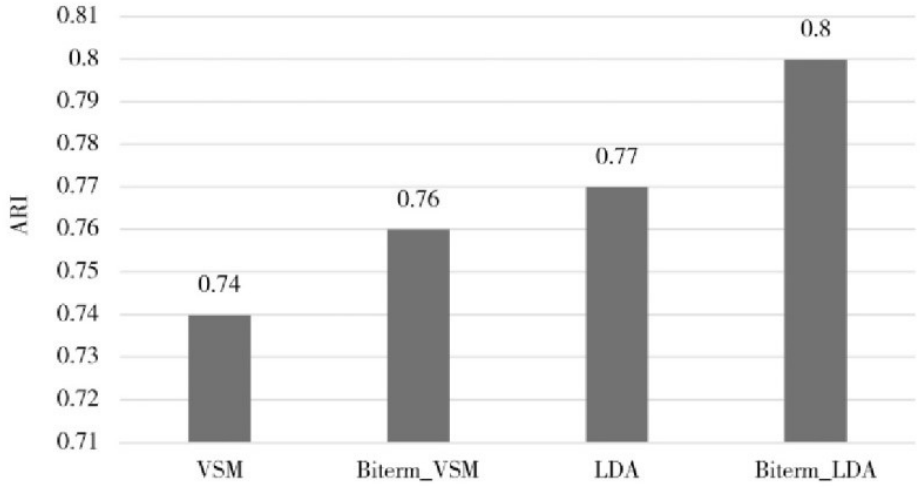

图 18 不同模型下的聚类性能对比

从图中我们可以 发现特征融合模型(Biterm__LDA)与其他模型(VSM、Biterm__VSM 和 LDA)的评价指标对比情况[8]。如果我们结合了 Biterm__VSM 模型和 LDA 模型的 Biterm__LDA 模型比单纯的 VSM 的性能要优；如果单独使用 Biterm__VSM 模型和 LDA 模型性能差不多；相比于前 3 个模型，本文提出的特征融合模型的性能是最优的。

5.7.2 实验 2

实验 2 的目的是为了验证主题为不同 K 值况下采用本文方法的对比情况，实验结果如图 19 所示。

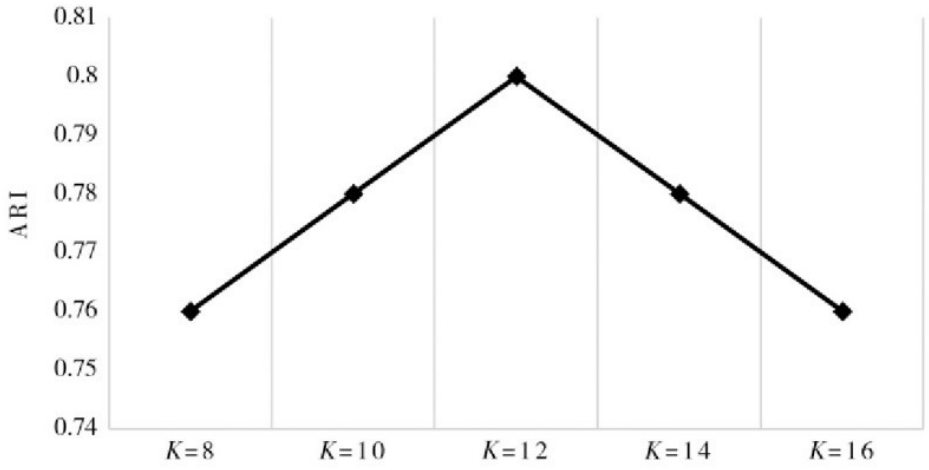

图 19 不同 k 值下的实验结果

通过图 19 我们可以观察出，当主题的数目越接近真实的留言内容时，效果越来越好；当主题的数目超过留言内容的时侯，效果越来越差。同时，我们也可以发现当 K=12 时，得到的聚类结果是最好的，而且和实际的热点留言内容是最接近的，说明本文提出的模型是有效的。

5.7.3 实验 3

实验 3 是为了验证 K-means 算法与密度聚类的 DBCAN 算法、谱聚类算法和凝聚式层次聚类算法 3 种算法之间的对比实验，并且是在主题的数目 K=12 的情况下进行的。实验结果如图 20 所示。

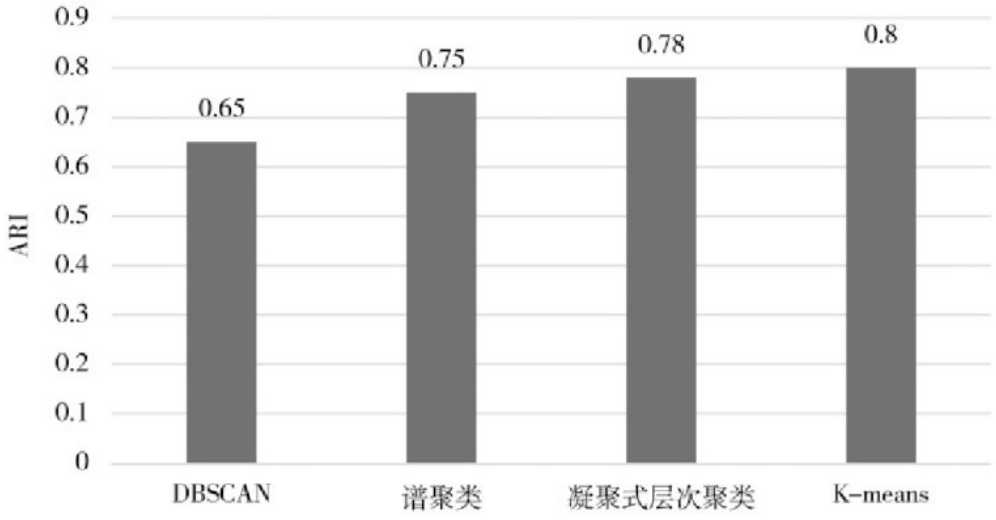

图 20　不同聚类算法的实验结果

六、 任务三

6.1 基于自然语言处理对答复意见的量化处理

基于问题一对文本进行处理的方式将答复意见的文本内容进行如下处理：

（1） 文档切分：将附件 4 中的答复意见文本内容单独存放到一个文件中；
（2） 文本分词：对新的文件中的文本内容进行词典的构造和分词算法的操作；
（3） 去停用词：将可省略的词语去除；
（4） 文本特征提取和词频统计：将文本出现的关键词进行提取并做词频统计；
（5） 文本向量化：用数学上的多维特征向量来表示一个文本；

6.2 评价答复意见与留言问题内容的相关性

我们如果要判断两个文章内容的相关性，这时候我们需要对数字映射后的特征做一个余弦相似度的匹配：即 a.dot(b) / sqrt(a^2 + b^2)

余弦相似度是通过两个向量之间的夹角来衡量向量相似性。给定两个向量 A 和 B，余弦相似性 θ 由向量的点积和长度决定，如下所示：

$$\cos(\theta) = \frac{A \bullet B}{\| A \| \| B \|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

同时要对两个文本中的关键词依次对应做相关系数的计算。从而通过余弦相似度和相关性系数的等级来进行评价留言详情和答复意见的相关性。

相关性系数分类大致如下：

0.8-1.0 极强相关
0.6-0.8 强相关
0.4-0.6 中等程度相关
0.2-0.4 弱相关
0.0-0.2 极弱相关或无相关

6.2.1 留言和答复文本预处理及余弦相似度的计算

在预处理过程中，主要分为以下步骤：

（1） 首先将留言详情和答复详情分别放到两个文件，由于 glob 模块读取的

是.txt 文本文件，因此分别存为留言详情.txt 和答复意见.txt.以便直接读取.txt 格式的作业。

（2） 过滤无效的字符串，为了降低重复程度，尽可能呈现准确的结果，要过滤掉空格、逗号、句号、双引号等无效字符。

（3） 根据留言详情和答复意见的内容，去停用词。去掉如“的”、“了”等字；使文本简化。

（4） 运用 jieba 分词对去除字符后的留言详情跟意见答复进行分词处理，将分词后的文本重新保存到一个新的.txt 文件中。合并后存为 key_word。

（5） 利用 python 编程实现相关性的处理得到两个向量后，套用余弦函数，计算两个向量的余弦相似度 similarity。

6.2.2 对相关性进行分析

由文本的预处理可得到留言主题和答复详情的关键词，（以附件四第一行信息为例）在留言详情中共提取出 91 个关键词，答复意见中共提取出 46 个关键词。进而建立相关指标表一：

留言详情（X）	X1：小区；X2:物业公司；X3:水电；X4:收费；X5：投票.....
答复详情（Y）	Y1：业委会；Y2:问题；Y3：物业公司；Y4：召开；Y5:标准；Y6:管理费……

再通过 MATLAB 利用计算相关系数（r）的关系式

$$r(X_i,Y_j)=\frac{Cov(X_i,Y_j)}{\sqrt{Var[X_i]Var[Y_j]}}$$

对相关指标表一进行处理。可得到类似表二相关系数矩阵

	X1	X2	X3	X4	X5	X_i
Y1	0.89	0.90	0.56	0.86	0.45	
Y2	0.65	0.85	0.90	0.90	0.20	
Y3	0.88	1.00	0.95	0.91	0.19	
Y4	0.50	0.23	0.10	0.34	0.10	
Y5	0.15	0.89	0.97	0.99	0.09	
Y6	0.89	0.92	0.98	1.00	0.15	
Y_j						

从而得到各向量指标的相关系数，经实验可评价出答复意见与留言详情的相关性的大小。

6.3 评价答复意见内容的完整性

6.3.1 制定答复意见的规范模式

首先对于答复意见的文本先要指定一个答题规范模式，例如：问候--留言问题复述--与该问题相关的部门和措施--目前状况--将给予的措施--结尾问候及答谢。根据多种类型的问题，从地方、部门、相应措施等方面建立多种答复规范模型。能够更精确的评价答复意见的完整性。

6.3.2 建立评价答复意见完整性的模型

关于评价答复意见的完整性模型，主要步骤包括：

（1） 对答复意见和留言详情进行文本预处理；
（2） 得到两部分相应的关键词，进行分别整合；
（3） 再通过留言详情与答复意见的相关性模型计算关键词所构建起来的相关性；
（4） 相关性小的可直接视为不完整答复，而计算出相关性大的答复意见则可进一步进入到答复规范模式进行检验。如果符合，则以完整答复的结果呈现，如不符合，则以不完整答复结果呈现。

其流程图可以用图 21 表示

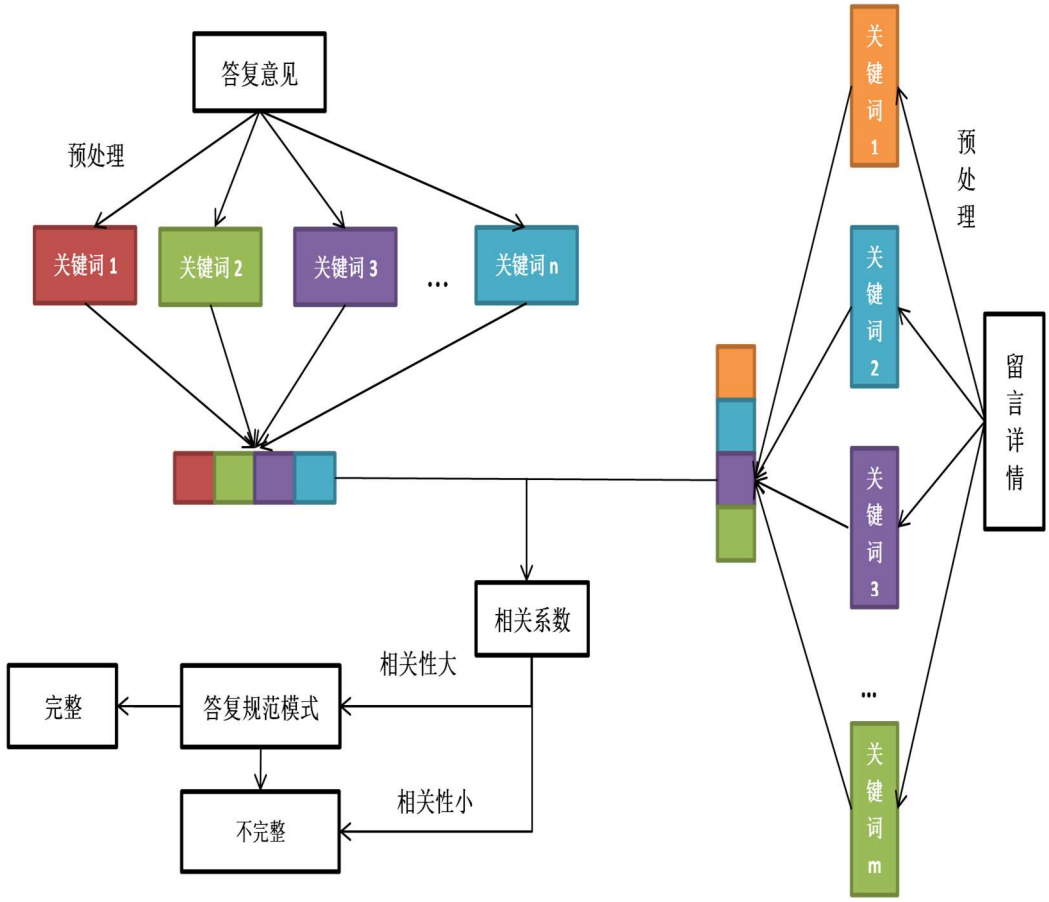

图 21 评价答复意见的完整性模型

6.4 评价答复意见内容的可解释性

6.4.1 可解释性的概念

在实现文本相关性和完整性的基础上，文本的可解释性也成为一个答复评价模型的关键，而对于答复意见内容的“可解释性”，可理解为文本本身为了让人顺利解读所应具备的必要条件。一个文本，小到一个句子，大至一篇文章，如果不具备这个必要条件，就很难达到传达意义的目的。也很难确保答复意见相应回答了留言问题和是否对问题的解决提供了清晰的答复与措施。

6.4.2 基于答复、主题、方面方法的可解释性研究

随着微信、微博、信箱等网络问政平台逐步成为了解民意、凝聚民气、解决实际生活问题的重要渠道，同时留言的爆炸性增长，人们在留言评论时会产生各种各样不同的问题，因此利用文本进行可解释性成为当下留言和答复系统领域的研究热潮。具体步骤如下：

（1） 对留言详情进行预处理，提取主题。并实现与答复意见的相关性；

（2） 实现相关性性后获到相应的答复意见，再对答复意见的完整性引用相应的指标进行评估；

（3） 基于答复的主题方法对评价模型的可解释性做出评价，主要通过主题的相关性进行评估；

（4） 再者基于主题的方面方法对文本进行划分，得到答复方向、相关部门对该问题的举措、解决该问题的日后措施等板块。进行再次评估。最终实现答复意见是否具有可解释性。

6.5 总结评价模型的构建如图 22 所示

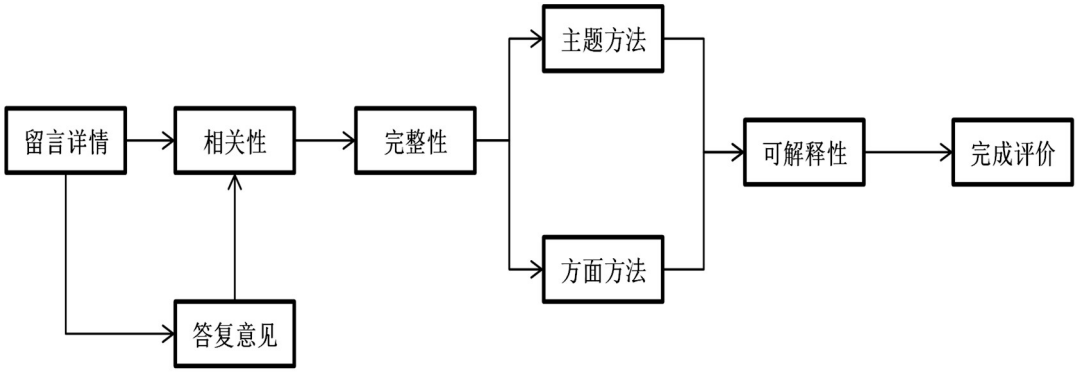

图 22 评价模型的构建

七、参考文献

[1]杨锐,陈伟,何涛,张敏,李蕊伶,岳芳.融合主题信息的卷积神经网络文本分类方法研究[J].现代情报,2020,40(04):42-49.

[2] 曾凡锋,李玉珂,肖珂.基于卷积神经网络的语句级新闻分类算法[J].计算机工程与设计,2020,41(04):978-982.DOI:10.16208/j.issn1000-7024.2020.04.013.

[3]陈欢,黄勃,朱翌民,俞雷,余宇新.结合 LDA 与 Self-Attention 的短文本情感分类方法[J/OL].计算机工程与应用:1-8.

[4]赖文辉,乔宇鹏.基于词向量和卷积神经网络的垃圾短信识别方法[J].计算机应用,2018,38(09):2469-2476.

[5] SRIVASTAVA N，HINTON G，KRIZHEVSKY A,et al.Dropout:A simple way to prevent neural networks from overfitting[J].The Journal of Machine Learning Research,2014,15(1):1929-1958.

[6] 洪伟俊,徐守余.基于 F-score 与支持向量机的砂体静态连通性定量评价[C].2018 油气田勘探与开发国际会议（IFEDC 2018）论文集.2018 油气田勘探与开发国际会议（IFEDC 2018）论文集.西安华线网络信息服务有限公司,2018:2158-2164.

[7] 李园园.基于 LDA 的游客在线评论主题分类——秦始皇兵马俑博物馆为例[J].河北企业，2020，（04）：43-44.

[8] 李海磊,杨文忠,李东昊,温杰彬,钱芸芸.基于特征融合的 K-means 微博话题发现模型[J].电子技术应用,2020,46(04):24-28+33.DOI:10.16157/j.issn.0258-7998.191367.

[9]张智慧，吴钰，杨福军.基于模型结构和事件日志的流程相似度计算[J].计算机测量与控制，2020,28(03):235-241.DOI:10.16526/j.cnki.11-4762/tp.2020.03.049.