

第八届“泰迪杯”

全国大学生数据挖掘挑战赛

论

文

作品名称：“智慧政务”中的文本挖掘应用（C 题）

## 摘 要

本文基于给出的来自互联网公开来源的群众问政留言记录，及相关部门的答复意见。利用自然语言和文本挖掘的方法一共解决了三个问题：1、群众留言分类；2、热点挖掘问题；3、答复意见评价。

对于问题 1：首先需要对附件 2 的群众留言进行数据预处理去停用词选取关键词，对群众留言的数据的特征有一个清晰的了解。首先利用 Jieba 分词工具对留言进行分词、词性标注、去除停用词，提取含有名词类的评论，运用 Python 绘制词云图获得词语的词频和权重数，然后构建一级分类标签模型，接着使用 sklearn 中的 chi2 方法来检验数据的拟合度和关联度，最后使用 SVM 分类算法来建立分类模型以及利用 F-Score 对分类模型进行评价。

对于问题 2：主要任务是热点信息挖掘，对网民提出的问题，我们将留言进行分词，并将需要在其中寻找答案的文本构成问答数据库，进行比较，找出文本相似度较高的留言。为此，我们从大量的数据中，把含有特定地点，人物的留言归为一类问题，在通过热度评价指标。排序出最热点的留言问题。我们使用 word2vec 构建数值空间，将词转化成机器能识别的数据，进而找出词与词之间的相似性和某些联系。以关键词为基础，建立 LDA 主题模型，进行文本分类，分析所有留言中的热点问题。再寻找出最优主题数，对主题进行评价。

对于问题 3：首先做的根据附件 4 相关部门对留言进行的答复意见，首先利用 python 对文件进行数据提取，然后利用自然语言处理工具 jieba 对留言进行数据预分词以及去停用词，也即数据预处理，然后对数据提取关键字，并进行留言详情和答复意见的关键字的匹配，进而再利用 TF-IDF 模型对处理过的数据先进行相关性的分析、在满足相关性的条件下进行完整性、以及可行性等角度的分析，将此数据化后，再对答复意见的质量给出一套评价方案并建立相关的模型。

**关键词：**TF-IDF；F-Score；word2vec；jieba；数据挖掘

## Abstract

This paper is based on the records of political messages from open sources of the Internet, as well as the responses of relevant departments. Three problems have been solved by using natural language and text mining methods: 1. Classification of mass messages; 2. Hot spot mining questions; 3. Evaluation of response opinions.

For question 1: first of all, we need to pre-process the public messages in Annex 2 to select keywords. Have a clear understanding of the characteristics of the data left by the masses. First of all, we use Jieba sub-company tools to advance the message. Line segmentation, part-of-speech tagging, removal of deactivated words, extraction of comments containing noun classes, and use Python to draw words. The cloud image obtains the word frequency and weight number of words, then constructs a first-level classification label model, and then uses sklearn. The chi2 method is used to test the fitting degree and correlation degree of the data. Finally, the SVM classification algorithm is used to establish the score. Class model and F-Score are used to evaluate the classification model.

For question 2: the main task is hot information mining, for the questions raised by netizens, we will divide the messages. Words, and will need to find answers in the text to form a question and answer database, compare, find out the text is similar, A higher degree message. For this reason, from a large number of data, we classify the messages containing specific locations and characters into a category of problems, through the heat evaluation index. Sort out the hottest message questions. We use word2vec to construct numerical space, convert words into data that can be recognized by machine and then find out the similarity and some connections between words. Based on keywords, LDA topic model is established, text classification is carried out, and the hot issues in all messages are analyzed. Then find out the optimal number of topics and evaluate the topics.

With regard to question 3: the first comments are based on the responses of the relevant departments in annex 4 to the messages. Firstly, python is used to extract the data from the file, and then the natural language processing tool jieba is used to presegment and deactivate the phonological words, that is, the data preprocessing, and then the keywords are extracted from the data, and the details of the messages are matched with the keywords of each complex opinion, and then the advanced integration and feasibility of the processed numbers are analyzed by using the TP TDP polar pair. Analysis of row correlation and completeness under the condition of satisfying correlation. After this effect is selected, Yang sets up the relevant model for the evaluation of the quality you see.

**Key words:** TF- IDF; F — Score; word2vec; jieba; Data Mining

# 目 录

1 绪论 .....	1
1.1 引言 .....	1
1.2 问题分析 .....	1
1.3 分析方法与过程 .....	2
2 数据预处理 .....	2
2.1 数据去重 .....	2
2.2 数据清洗 .....	3
3 问题一：群众留言分类 .....	4
3.1 留言分词 .....	4
3.2 构建一级标签分类模型 .....	8
3.3 模型的评估 .....	11
4 问题二：热点问题挖掘 .....	14
4.1 LDA 主题模型 .....	15
4.2 寻找最优主题数 .....	17
4.3 主题评价结果 .....	19
5. 问题三：答复意见评价 .....	20
5.1 问题分析 .....	20
5.2 思路分析 .....	20
5.3 数据预处理 .....	23
5.4 关键词匹配 .....	25
5.5 模型设计 .....	26

参考文献 .....	28
------------	----

# 1 绪论

## 1.1 引言

随着互联网的高速发展，信息量成指数增长，越来越多的信息处理只能借助计算机，而人工处理也逐将被淘汰。互联网产生的海量数据中蕴含着大量的信息，已成为政府和企业的一个重要数据来源。人工分类的精度高，质量好，但效率较低，在这个大数据时代，已经不能满足需求了。我们要借助各种数据分析软件，来处理海量数据，进行推测，预知结果，挖掘大量重要信息。当发生重大社会事件时，网络问政平台发挥了作为公众讨论和发表意见的平台的重要性。各类与社情民意相关的留言数量不断攀升，划分和热点整理的工作可以减轻相关部门的负担，同时提高办事效率。在某一时间段内多次出现的信息或出现次数较多的信息称为热点问题。我们要从庞大的留言信息中及时找出热点问题，帮助相关部门针对性的处理，提高服务效率，解决民生问题，增强民心。本文利用自然语言处理(NLP)和文本挖掘方法，对文本进行分类，找出热点问题，并进行热点排序。

处理文本分类主要应用于信息检索，机器翻译，信息过滤和邮件分类的任务。从1960年的人工和关键词的分类方式，到现在才用词向量来进行文本分类，文本分类依旧是社会生活中信息邮件分类的主要任务方式。按照一定的划分体系对留言进行分类（三级标签体系），以便后续将群众留言分派至相应的职能部门。

通过自然语言处理方法，首先将文本预处理，分词，取出停用词，过滤低频词。再使文本向量化，使用向量空间模型或概率统计模型对文本进行表示，使计算机能够理解计算。接着是文本特征提取和选择，特征项的选择和特征权重的计算是文本分类的核心。最后选择分类器和建立回归模型。

基于对文本分类的理解和认识，本文将立足于背景和问题，构建词袋模型和语义匹配度模型，<sup>[1]</sup>完成网络问政平台上留言问题的分类和挖掘。在完成对题目所给问题集的数据分析已经预处理工作之后，在语义，相似度，分类上都表现出优越的效果。因此自然语言处理在文本分类有着积极的作用。

## 1.2 问题分析

针对问题1，实际上这是一个文本分类问题，首先需要对附件2的群众留言进行数据预处理，选取关键词，对群众留言的数据的特征有一个清晰的了解，首先采用CHI的方法对群众留言进行特征提取，接着使用TF-IDF来计算网络问政平台中群众留言有重要作用的词语的权重，实现文本关键词的抽取，然后使用SVM分类算法

来建立分类模型，最后使用 F-Score 对分类模型进行评价。

针对问题 2:群众留言的分类问题，先进行分类标签的定义以及数据的简单清洗，然后进行数据的读取与抽取。采用当前广泛使用的基于 Python 的中文分词工具 jieba 对文本进行分词，去停用词来降低句子噪音对句子的理解，减少特征词的数量。根据词云图结果来修改停用词的去除来提炼数据，提高数据的有效性。分类方法是利用 sklearn 实现多分类 demo，引入相应的库，进行加载数据及数据格式转化，建立训练模型，最后进行相应的性能评估。

针对问题 2: 热点问题挖掘，先对数据进行简单的清洗，可以先将留言编号和留言用户以及留言时间这三列无用数据删除，然后进行数据的读取与抽取。绘制词云图，再用 jieba 对文本进行分词，从而找到热点问题。找到热点问题后，热度指数可以用该问题出现的次数来评价，然后将热点问题进行排列。

针对问题 3: 答复意见的评价，从答复的相关性、完整性、可解释性等角度指出答复意见的不足并给出相关的建议。

### 1.3 分析方法与过程

在文本挖掘的过程中，先进行数据抽取和清洗将无关数据删除，再进行分词去停用词等操作可去掉很多无太大意义的词留下特征性的词，得出处理后的数据，建立文本分类的模型，再将数据进行实践，最后对所建的模型进行评价和优化。如图 1 所示。

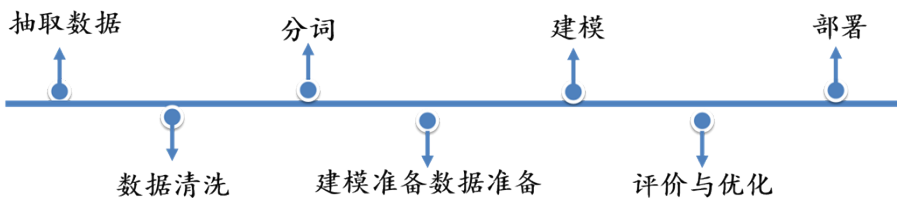


图 1 挖掘流程

## 2 数据预处理

高质量的数据集是模型匹配和优化的基础，对整个数据集进行分析处理可以促进对数据集的全面认知，从而更好地对数据进行处理，进一步提高数据集的质量。根据分析结果，更容易选择预处理阶段的相关参数，减少重复摸索的概率。

### 2.1 数据去重

数据产生的渠道越来越多，速度越来越快，大量的数据为数据分析和处理带来

了较大的难度，数据量剧增会导致数据的可靠性不足，为处理数据之间的关系，降低冗余数据，因此首先做数据去重。以下为数据去重的代码：

```
import pandas as pd

import re

import jieba.posseg as psg

import numpy as np

# 去重，去除完全重复的数据

reviews=pd.read_excel("C:/chapter12/demo/tmp/f2.xlsx")

len(reviews)      #9210

reviews=reviews[['留言主题','一级标签']].drop_duplicates()

len(reviews)      #8908

content_topic=reviews['content_topic']

print(reviews)
```

通过数据去重得到的结果如图 2 所示：

	contentid	content_type	
0	A市西湖建筑集团占道施工有安全隐患	城乡建设	
1	A市在水一方大厦人为烂尾多年，安全隐患严重	城乡建设	
2	投诉A市A1区苑物业违规收停车费	城乡建设	
3	A1区蔡锷南路A2区华庭楼顶水箱长年不洗	城乡建设	
4	A1区A2区华庭自来水好大一股霉味	城乡建设	
5	投诉A市盛世耀凯小区物业无故停水	城乡建设	
6	咨询A市楼盘集中供暖一事	城乡建设	
7	A3区桐梓坡西路可可小城长期停水得不到解决	城乡建设	
8	反映C4市收取城市垃圾处理费不平等的问题	城乡建设	
9	A3区魏家坡小区脏乱差	城乡建设	
10	A市魏家坡小区脏乱差	城乡建设	
11	A2区泰华一村小区第四届非业委会涉嫌侵占小区业主公共资金	城乡建设	
12	A3区梅溪湖壹号御湾业主用水难	城乡建设	
13	A4区鸿涛翡翠湾强行对入住的业主关水限电	城乡建设	
14	地铁5号线施工导致A市锦楚国际星城小区三期一个月停电10来次	城乡建设	
15	A6区润和紫郡用电的问题能不能解决	城乡建设	
16	A市锦楚国际新城从6月份开始停电好多次了	城乡建设	
17	给A9市城区南西片区城铁站设立的建议	城乡建设	
18	请A6区政府加大对滨水新城的绿化建设	城乡建设	
19	A5区楚府线几个小区经常停电	城乡建设	
20	请调查西地省建望集团及西地省辉东安建工程有限公司的违法行为	城乡建设	

图 2 数据去重得到结果

## 2.2 数据清洗

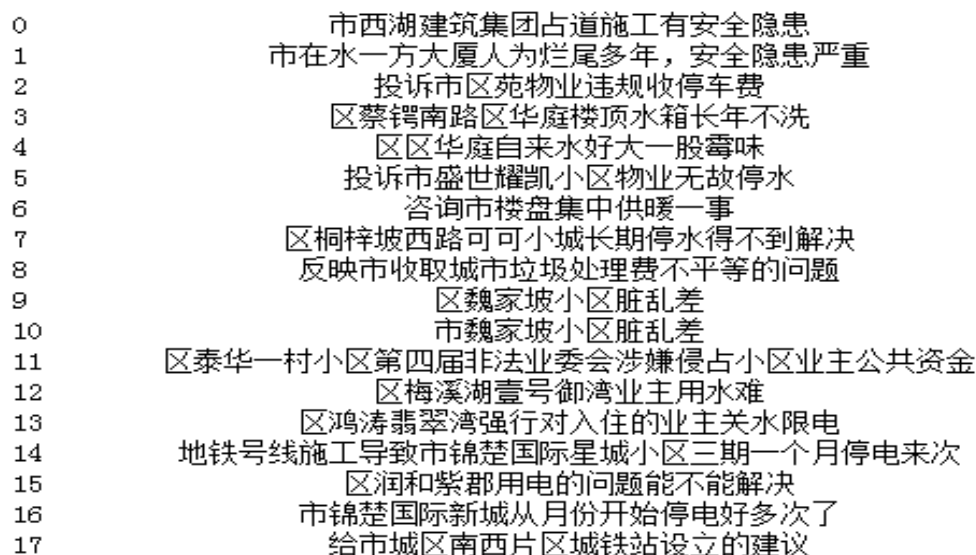
群众留言内容包含许多停用词、中英文符号及数字等与主题无关的内容, 通过数据清洗可以减少文章中无效内容，避免对模型造成干扰，有助于提升分类模型的准确率。使用的停用词表主要来自于网络. 上常用的中英文停用词和后期自己添加的停



用词。以下为去除英文、数字等停用词的代码<sup>[2]</sup>：

```
# 去除英文、数字等
strinfo = re.compile('[0-9a-zA-Z]| ')
content_id = content_topic.apply(lambda x: strinfo.sub("", x))
print(content_topic)
```

通过去除英文、数字等处理后，得到的结果如图 3 所示：



0	市西湖建筑集团占道施工有安全隐患
1	市在水一方大厦人为烂尾多年，安全隐患严重
2	投诉市区苑物业违规收停车费
3	区蔡锷南路区华庭楼顶水箱长年不洗
4	区区华庭自来水好大一股霉味
5	投诉市盛世耀凯小区物业无故停水
6	咨询市楼盘集中供暖一事
7	区桐梓坡西路可可小城长期停水得不到解决
8	反映市收取城市垃圾处理费不平等的问题
9	区魏家坡小区脏乱差
10	市魏家坡小区脏乱差
11	区泰华一村小区第四届非法业委会涉嫌侵占小区业主公共资金
12	区梅溪湖壹号御湾业主用水难
13	区鸿涛翡翠湾强行对入住的业主关水限电
14	地铁号线施工导致市锦楚国际星城小区三期一个月停电来次
15	区润和紫郡用电的问题能不能解决
16	市锦楚国际新城从月份开始停电好多次了
17	给市城区南西片区城铁站设立的建议

图 3 去除英文、数字等后的结果

### 3 问题一：群众留言分类

针对问题 1，实际上这是一个文本分类问题，首先需要对附件 2 的群众留言进行数据预处理，选取关键词，对群众留言的数据的特征有一个清晰的了解。首先利用 Jieba 分词工具对留言进行分词、词性标注、去除停用词，提取含有名词类的评论，运用 Python 绘制词云图获得词语的词频和权重数，然后构建一级分类标签模型，接着使用 sklearn 中的 chi2 方法来检验数据的拟合度和关联度，最后利用 F-Score 对分类模型进行评价<sup>[3]</sup>。

#### 3.1 留言分词

文本分词的目的是将原始文本以空格为分隔符按词进行拆分，主要分词算法有基于字符串匹配的正或逆向最大匹配，基于理解的句法和语义分析等<sup>[1]</sup>。当前国内比较流行的开源中文分词工具主要有 Jieba、SnowNIP、THULAC、NLPIR 等。Jieba 是目前使用人数最多的中文分词软件，根据应用场景的不同可以采用其中不同的分词模式，还可以通过加载自定义词典，提高对特定领域词语识别的准确率，同时，

由于其利用了 HMM 模型和 Viterbi 算法，对新词识别的效果也较好。

### 3.1.1 分词、词性标注、去除停用词

停用词是指在信息检索中，为节省存储空间和提高搜索效率，在处理自然语言数据（或文本）之前或之后会自动过滤掉某些字或词，这些字或词即被称为 Stop Words（停用词）。这些停用词都是人工输入，非自动化生成的，生成后的停用词会形成一个停用词表，停用词<sup>[2]</sup>主要包括英文字符、数字、数学字符、标点符号及使用频率特高的单汉字等，去除停用词可以有效地降低文本噪声。进行分词、词性标注、去除停用词代码如下图 4：

```
# C-3 分词、词性标注、去除停用词代码

# 分词
worker = lambda s: [(x.word, x.flag) for x in posseg.cut(s)] # 自定义简单分词函数
seg_word = content_topic.apply(worker)

# 将词语转为数据框形式，一列是词，一列是词语所在的句子ID，最后一列是词语在该句子的位置
n_word = seg_word.apply(lambda x: len(x)) # 每一评论中词的个数

n_content = [[x+1]*y for x,y in zip(list(seg_word.index), list(n_word))]
index_content = sum(n_content, []) # 将嵌套的列表展开，作为词所在评论的id

seg_word = sum(seg_word, [])
word = [x[0] for x in seg_word] # 词

nature = [x[1] for x in seg_word] # 词性

content_type = [[x]*y for x,y in zip(list(reviews['content_type']), list(n_word))]
content_type = sum(content_type, []) # 评论类型

result = pd.DataFrame({"index_content": index_content,
                       "word": word,
                       "nature": nature,
                       "content_type": content_type})

# 删除标点符号
result = result[result['nature'] != 'x'] # x表示标点符号

# 删除停用词
stop_path = open("C:/chapter12/demo/data/hlt_stop_words.txt", 'r', encoding='UTF-8')
stop = stop_path.readlines()
stop = [x.replace('\n', '') for x in stop]
word = list(set(word) - set(stop))
result = result[result['word'].isin(word)]

# 构造各词在对应评论的位置列
n_word = list(result.groupby(by = ['index_content'])['index_content'].count())
index_word = [list(np.arange(0, y)) for y in n_word]
index_word = sum(index_word, []) # 表示词语在改评论的位置

# 合并评论id，评论中词的id，词，词性，评论类型
result['index_word'] = index_word
```

图 4 分词、词性标注、去除停用词代码

通过进行分词、词性标注、去除停用词代码处理后，得到的结果如图 5 所示：

	content_type	index_content	nature	word	index_word
0	城乡建设	1	n	市	0
1	城乡建设	1	ns	西湖	1
2	城乡建设	1	n	建筑	2
3	城乡建设	1	n	集团	3
4	城乡建设	1	v	占	4
5	城乡建设	1	q	道	5
6	城乡建设	1	vn	施工	6
8	城乡建设	1	i	安全隐患	7
9	城乡建设	2	n	市	0
10	城乡建设	2	i	在水一方	1

图 5 分词、词性标注、去除停用词代码结果

### 3.1.2 提取含名词的评论

为绘制词云图，能够准确快速地筛选出重要文本信息，把关键字以图片的形式展现出来，帮助提取群众留言中的关键词，提取含有名词类的评论。提取含有名词类的评论代码如下图 5：

```
# C-4 提取含有名词的评论

# 提取含有名词类的评论
ind = result[['n' in x for x in result['nature']]]['index_content'].unique()
result = result[[x in ind for x in result['index_content']]]
```

提取含名词的评论得到的结果如图 6 所示：

	content_type	index_content	nature	word	index_word
0	城乡建设	1	n	市	0
1	城乡建设	1	ns	西湖	1
2	城乡建设	1	n	建筑	2
3	城乡建设	1	n	集团	3
4	城乡建设	1	v	占	4
5	城乡建设	1	q	道	5
6	城乡建设	1	vn	施工	6
8	城乡建设	1	i	安全隐患	7

图 6 群众留言中提取含名称评论结果

### 3.1.3 绘制词云查看分词效果

词云图，也叫文字云，是对文本中出现频率较高的“关键词”予以视觉化的展现，词云图过滤掉大量的低频低质的文本信息，使得浏览者只要一眼扫过文本就可领略文本的主旨，进而获得词语的词频和权重数。绘制词云图代码如下：

```
import matplotlib.pyplot as plt
from wordcloud import WordCloud
frequencies = result.groupby(by = ['word'])['word'].count()
frequencies = frequencies.sort_values(ascending = False)
background_Image=plt.imread('C:/chapter12/demo/data/pl.jpg')
wordcloud = WordCloud(font_path="simfang.ttf",
                        max_words=100,
                        background_color='white',
                        mask=background_Image)
my_wordcloud = wordcloud.fit_words(frequencies)
plt.imshow(my_wordcloud)
plt.axis('off')
plt.show()
```

绘制词云图结果如图 7 所示：



例如我们前面已经转换好的 tf-idf 的 features。

当我们有了词向量以后我们就可以开始训练我们的分类器。分类器训练完成后, 就可以对没有见过的 review 进行预测。

## 2. 朴素贝叶斯分类器

朴素贝叶斯分类器最适合用于基于词频的高维数据分类器, 最典型的应用如垃圾邮件分类器等, 准确率可以高达 95%以上。这里我们使用的是 sklearn 的朴素贝叶斯分类器 MultinomialNB, 我们首先将 review 转换成词频向量, 然后将词频向量再转换成 TF-IDF 向量, 还有一种简化的方式是直接使用 TfidfVectorizer 来生成 TF-IDF 向量(正如前面生成 features 的过程), 这里我们还是按照一般的方式将生成 TF-IDF 向量分成两个步骤: 1. 生成词频向量。 2. 生成 TF-IDF 向量。最后我们开始训练我们的 MultinomialNB 分类器。

```
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.naive_bayes import MultinomialNB

X_train, X_test, y_train, y_test = train_test_split(df['cut_review'],
df['cat_id'], random_state = 0)

count_vect = CountVectorizer()
X_train_counts = count_vect.fit_transform(X_train)

tfidf_transformer = TfidfTransformer()
X_train_tfidf = tfidf_transformer.fit_transform(X_train_counts)

clf = MultinomialNB().fit(X_train_tfidf, y_train)
```

当模型训练完成后, 我们让它预测一些自定义的 review 的分类。不过我们首先编写一个预测函数 myPredict

```
def myPredict(sec):
    format_sec=""
    for w in list(jb.cut(remove_punctuation(sec))) if w not in stopwords])
        pred_cat_id=clf.predict(count_vect.transform([format_sec]))
    print(id_to_cat[pred_cat_id[0]])
```

## 3. 模型的选择

接下来我们尝试不同的机器学习模型, 并评估它们的准确率, 我们将使用如下四种模型:

Logistic Regression(逻辑回归)

(Multinomial) Naive Bayes(多项式朴素贝叶斯)

```

Linear Support Vector Machine(线性支持向量机)
Random Forest(随机森林)
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.svm import LinearSVC
from sklearn.model_selection import cross_val_score

models = [
    RandomForestClassifier(n_estimators=200,                max_depth=3,
random_state=0),
    LinearSVC(),
    MultinomialNB(),
    LogisticRegression(random_state=0),
]
CV = 5
cv_df = pd.DataFrame(index=range(CV * len(models)))
entries = []
for model in models:
    model_name = model.__class__.__name__
    accuracies = cross_val_score(model, features, labels,
scoring='accuracy', cv=CV)
    for fold_idx, accuracy in enumerate(accuracies):
        entries.append((model_name, fold_idx, accuracy))
cv_df = pd.DataFrame(entries, columns=['model_name', 'fold_idx',
'accuracy'])

import seaborn as sns
sns.boxplot(x='model_name', y='accuracy', data=cv_df)
sns.stripplot(x='model_name', y='accuracy', data=cv_df,
              size=8, jitter=True, edgecolor="gray", linewidth=2)

plt.show()

```

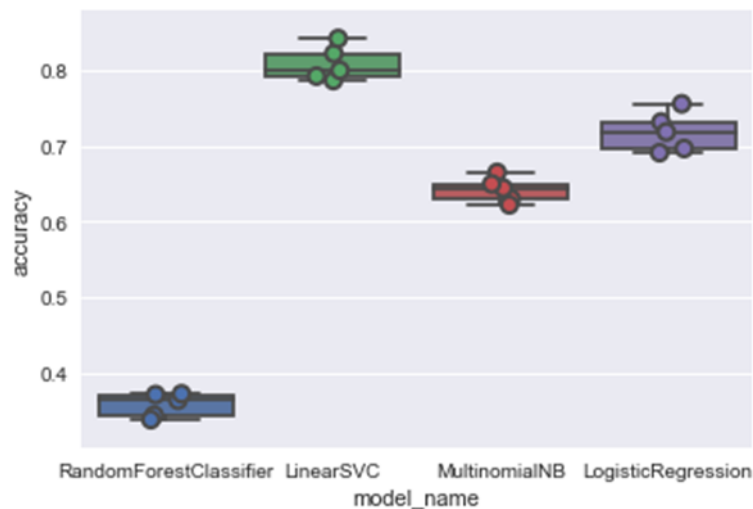


图 9 箱体图

从可以箱体图上可以看出随机森林分类器的准确率是最低的，因为随机森林属于集成分类器(有若干个子分类器组合而成)，一般来说集成分类器不适合处理高维数据(如文本数据)，因为文本数据有太多的特征值，使得集成分类器难以应付，另外三个分类器的平均准确率都在 80%以上。其中线性支持向量机的准确率最高。

```
cv_df.groupby('model_name').accuracy.mean()
```

### 3.3 模型的评估

#### 1. 混淆矩阵

针对平均准确率最高的 LinearSVC 模型，我们将查看混淆矩阵，并显示预测标签和实际标签之间的差异。

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, confusion_matrix

#训练模型
model = LinearSVC()
X_train, X_test, y_train, y_test, indices_train, indices_test = train_test_split(features, labels,
df.index,

test_size=0.33, stratify=labels, random_state=0)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

#生成混淆矩阵
conf_mat = confusion_matrix(y_test, y_pred)
fig, ax = plt.subplots(figsize=(10,8))
```



```
sns.heatmap(conf_mat, annot=True, fmt='d',
             xticklabels=cat_id_df.cat.values, yticklabels=cat_id_df.cat.values)
plt.ylabel('实际结果',fontsize=18)
plt.xlabel('预测结果',fontsize=18)
plt.show()
```

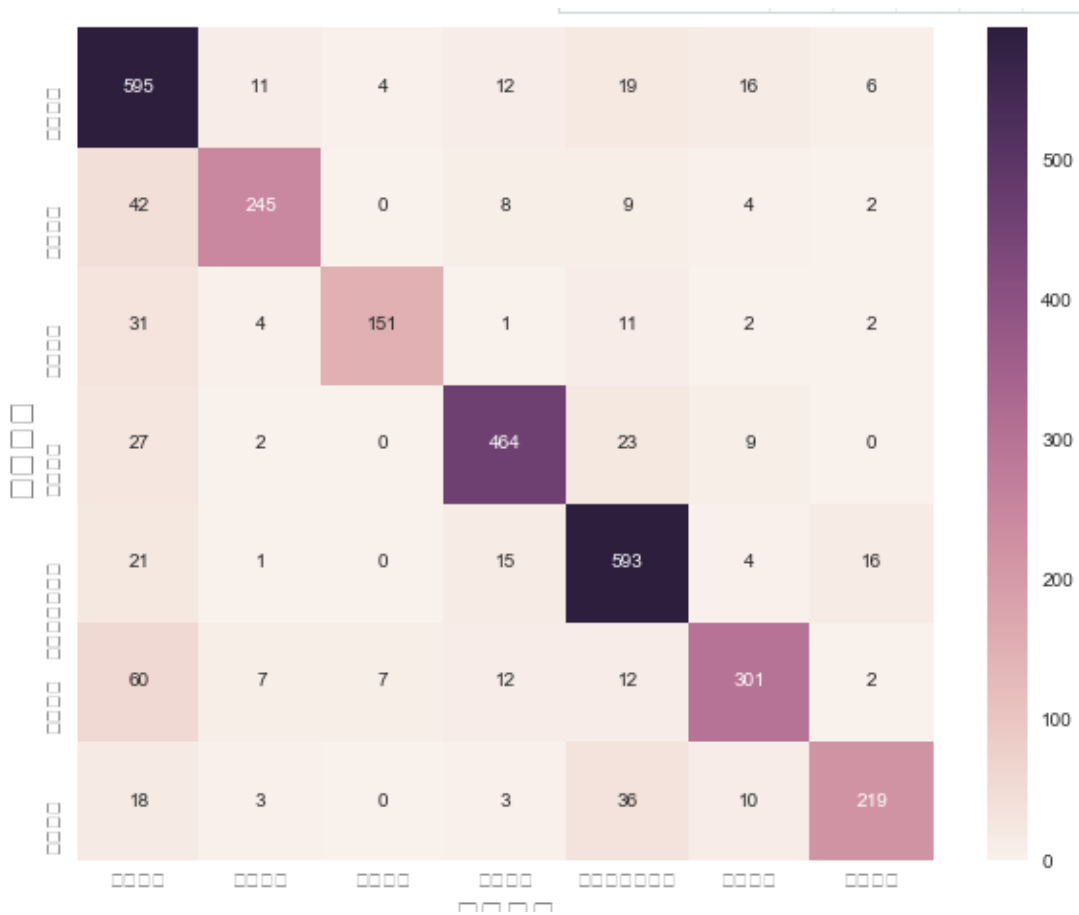


图 10 混淆矩阵

混淆矩阵的主对角线表示预测正确的数量,除主对角线外其余都是预测错误的数量.从上面的混淆矩阵可以看出"蒙牛"类预测最准确,只有一例预测错误。“平板”和“衣服”预测的错误数量教多。

多分类模型一般不使用准确率(accuracy)来评估模型的质量,因为 accuracy 不能反应出每一个分类的准确性,因为当训练数据不平衡(有的类数据很多,有的类数据很少)时, accuracy 不能反映出模型的实际预测精度,这时候我们就需要借助于 F1 分数、ROC 等指标来评估模型。

## 2. F1 分数 (F1 Score)

F1 分数 (F1 Score)，是统计学中用来衡量二分类（或多任务二分类）模型精确度的一种指标。它同时兼顾了分类模型的准确率和召回率。F1 分数可以看作是模型准确率和召回率的一种加权平均，它的最大值是 1，最小值是 0，值越大意味着模型越好。假如有 100 个样本，其中 1 个正样本，99 个负样本，如果模型的预测只输

出 0，那么正确率是 99%，这时候用正确率来衡量模型的好坏显然是不对的。

	真实 1	真实 0
预测 1	True Positive(TP)真阳性	False Positive(FP)假阳性
预测 0	False Negative(FN)假阴性	True Negative(TN)真阴性 <sup>[4]</sup>

查准率（precision），指的是预测值为 1 且真实值也为 1 的样本在预测值为 1 的所有样本中所占的比例。以西瓜问题为例，算法挑出来的西瓜中有多少比例是好西瓜。

$$p = \frac{TP}{TP + FP}$$

召回率（recall），也叫查全率，指的是预测值为 1 且真实值也为 1 的样本在真实值为 1 的所有样本中所占的比例。所有的好西瓜中有多少比例被算法挑了出来。

$$\text{召回率 } r = \frac{TP}{TP + FN}$$

F1 分数（F1-Score），又称为平衡 F 分数（BalancedScore），它被定义为精确率和召回率的调和平均数。

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

更一般的，我们定义  $F_\beta$  分数为：

$$F_\beta = (1 + \beta^2) \cdot \frac{\text{precision} \cdot \text{recall}}{(\beta^2 \cdot \text{precision}) + \text{recall}}$$

除了 F1 分数之外，F0.5 分数和 F2 分数，在统计学中也得到了大量应用，其中，F2 分数中，召回率的权重高于精确率。

查看各个类的 F1 分数代码如下：

```
from sklearn.metrics import classification_report
print('accuracy %s' % accuracy_score(y_pred, y_test))
print(classification_report(y_test, y_pred, target_names=cat_id_df['cat'].values))
```

各个类的 F1 分数结果如图 11 所示：

accuracy 0.844736842105					
	precision	recall	f1-score	support	
城乡建设	0.75	0.90	0.82	663	
环境保护	0.90	0.79	0.84	310	
交通运输	0.93	0.75	0.83	202	
教育文体	0.90	0.88	0.89	525	
劳动和社会保障	0.84	0.91	0.88	650	
商贸旅游	0.87	0.75	0.81	401	
卫生计生	0.89	0.76	0.82	289	
avg / total	0.85	0.84	0.84	3040	

图 11 各个类的 F1 分数结果

从以上 F1 分数上看，“教育文体”类的 F1 分数最大，“商贸旅游”类 F1 分数最低有 81%。

## 4 问题二：热点问题挖掘

网络问政平台作为传播民情信息的主要渠道，留言信息也很重要，在反映出热点问题时，也会看到更多民众对问题的关注度。目前定义文本信息热度方法有很多种：对关键特征词进行聚类分析，CURE 算法,箱线图识别法等，但都没有办法解决选用哪些特征作为评判热点留言标准的问题。为了从大量的留言数据中，寻找出热点问题，我们首先要识别出相似的留言，从大量的数据中，把含有特定地点，人物的留言归为一类问题，在通过热度评价指标。排序出最热点的留言问题。我们使用 word2vec 构建数值空间，将词转化成机器能识别的数据，进而找出词与词之间的相似性和某些联系。以关键词为基础，建立 LDA 主题模型，进行文本分类<sup>[5]</sup>，分析所有留言中的热点问题。再寻找出最优主题数，对主题进行评价。

通过 Word2vec 构建数值空间，输出神经网络的权重值。因为文本中每个词的权重是不同的，我们使用词权重的算法 TF-IDF 算法，该算法不仅考虑了每个词在句子中出现的频率，也考虑了词权重的算法<sup>[2]</sup>。TF-IDF 特征提取用 tfidf 权重计算方法构建文档向量空间。TF-IDF 的分数代表了词语在文档和整个语料库中的相对重要性

$TF(t) = (\text{该词语在文档出现的次数}) / (\text{文档中词语的总数})$

$IDF(t) = \log_e (\text{文档总数} / \text{出现该词语的文档总数})$

TF-IDF 向量可以由不同级别的分词产生（单个词语，词性，多个词（n-grams））

为了识别出表达多样化的人群和地点，识别相似留言，我们是用读独热表示 (one-hot representation) 实现自然语言处理数字化。

独热编码解决了分类器不好处理属性数据的问题，在一定程度上也起到了扩充特征的作用。它的值只有 0 和 1，不同的类型存储在垂直的空间。离散特征进行 one-hot 编码后，编码后的特征，其实每一维度的特征都可以看做是连续的特征。就可以跟对连续型特征的归一化方法一样，对每一维特征进行归一化。比如归一化到 [-1,1] 或归一化到均值为 0, 方差为 1。

建立 LDA 模型，找到每一个留言的主题分布和每一个主题中词的分。

LDA (Latent Dirichlet Allocation) 是一种文档主题生成模型<sup>[8]</sup>，也称为一个三层贝叶斯概率模型，包含词、主题和文档三层结构。如图 12 所示。

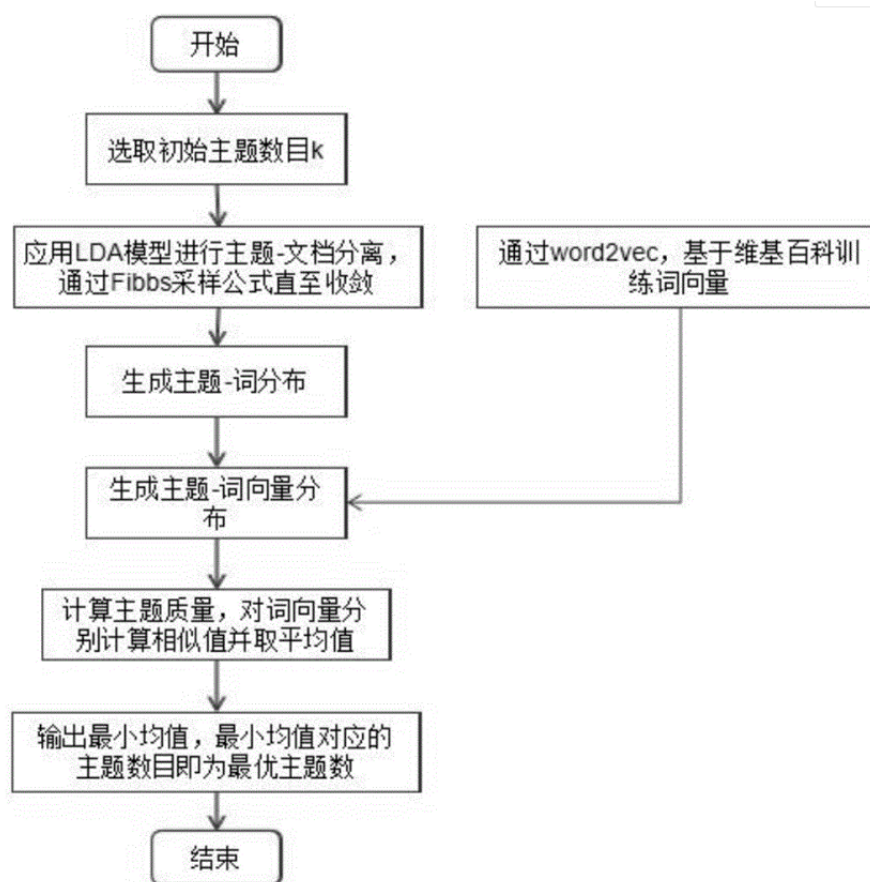


图 12 建立 LDA 模型步骤

## 4.1 LDA 主题模型

LDA 的目的就是要识别主题，即把文档—词汇矩阵变成文档—主题矩阵（分布）和主题—词汇矩阵（分布）。文档集合  $D$ ，主题（topic）集合  $T$ ，对每个  $D$  中的文档  $d$ ，对应到不同 Topic 的概率  $\theta_d = \langle p_{t1}, \dots, p_{tk} \rangle$ ，其中， $p_{ti}$  表示  $d$  对应  $T$  中第  $i$  个 topic 的概率。计算方法是直观的， $p_{ti} = n_{ti} / n$ ，其中  $n_{ti}$  表示  $d$  中对应第  $i$  个 topic 的词的数量， $n$  是  $d$  中所有词的总数。

对每个 T 中的 topic<sub>t</sub>，生成不同单词的概率  $\phi_t = \langle \phi_{t1}, \dots, \phi_{tV} \rangle$ ，其中， $\phi_{ti}$  表示 t 生成 VOC 中第 i 个单词的概率。计算方法同样很直观， $\phi_{ti} = N_{ti} / N_t$ ，其中  $N_{ti}$  表示对应到 topic<sub>t</sub> 的 VOC 中第 i 个单词的数目， $N_t$  表示所有对应到 topic<sub>t</sub> 的单词总数。<sup>[9]</sup> 如图 13 所示。

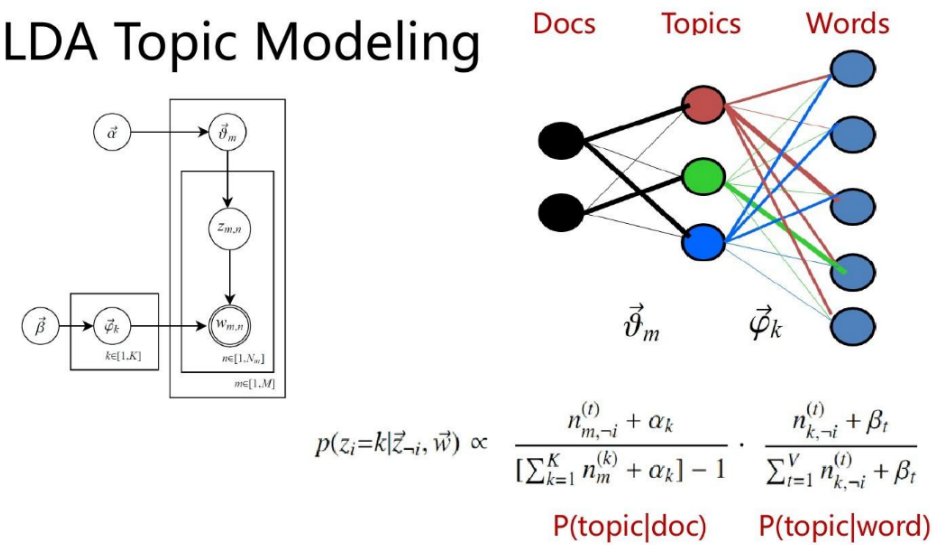


图 13LDA 模型原理

假设事先给定了这几个主题：Arts、Budgets、Children、Education，然后通过学习训练，获取每个主题 Topic 对应的词语。如图 14 所示。

“Arts”	“Budgets”	“Children”	“Education”
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

图 14 主题集合示意图

然后以一定的概率选取上述某个主题，再以一定的概率选取那个主题下的某个单词，不断的重复这两步，最终生成如下图所示的一篇文章。如图 15 所示。

The William Randolph Hearst Foundation will give \$1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be \$200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive \$400,000 each. The Juilliard School, where music and the performing arts are taught, will get \$250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual \$100,000 donation, too.

图 15 文档主题分布示意图

LDA 的核心公式如下：

$$p(w|d)=p(w|t)*p(t|d)$$

LDA 主题模型就是通过“文档——词语”矩阵进行训练，一定的概率推测出文档的主题。构造主题寻优函数如下：

```
def cos(vector1, vector2): # 余弦相似度函数
    dot_product = 0.0;
    normA = 0.0;
    normB = 0.0;
    for a,b in zip(vector1, vector2):
        dot_product += a*b
        normA += a**2
        normB += b**2
    if normA == 0.0 or normB==0.0:
        return(None)
    else:
        return(dot_product / ((normA*normB)**0.5))
```

## 4.2 寻找最优主题数

LDA 主题数量有多个，如何选取最优的主题，在主题与主题之间，主题与词语之间关联如何衡量？一种基于词汇相似性的 LDA 主题模型最优主题数确定方法，其特征在于，包括以下步骤<sup>[6]</sup>：

- 步骤 1：选取初始 k 值，作为 LDA 主题模型初始主题数目；
- 步骤 2：进行文档主题分离，采样主题，直至收敛；
- 步骤 3：生成主题-词分布，记为(T1, w11, w12, ..., w1n)、(T2, w21, w22, ..., w2n)、...、(Tn, wn1, wn2, ..., wnn)；其中，T1、T2、...、Tn 为 n 个主题，wij 为每个主题下的词分布；
- 步骤 4：将主题-词分布转换为主题-词向量分布；

步骤 5: 计算主题质量, 对每个主题下的词向量两两计算相似值, 获取平均值;

步骤 6: 绘制曲线, 为某个主题下的词语相似度平均值, Topic#为对应主题; 平均相似度达到最小时, 每个主题下的词分布倾向于表达一个主题, 分类模型达到最优。

作为优选, 步骤 2 中, 根据 Gibbs 采样公式采样主题。

作为优选, 步骤 4 中, 基于维基百科通过 word2vec 训练词向量, 将主题-词分布转换为主题-词向量分布。

作为优选, 步骤 5 中, 对每个主题下的词向量两两计算相似值, 计算方法是, 选取主题 T, 通过向量相加平均法得到每个主题下的主题词相似度之和的平均值, 其公式如下所示:

其中, NT 为主题数, w 为主题 T 下的主题词数目,  $e(w_i, w_j)$  为两词语间的相似度, 相似度通过余弦值得到, 即:

$w_i$  和  $w_j$  分别为词语的向量表示; 主题词 i 与主题词 j 计算相似度值, 然后取得主题 T 下所有分布词的相似度之和, 计算得到平均值。

作为优选, 步骤 6 中所述绘制曲线方法为: 为某个主题下的词语相似度平均值, 通过计算不同 Topic#下的值, 绘制出横坐标为 Topic#数, 纵坐标为的折线图, 基于连续的 Topic#数, 找到最小值点, 最小值点对应的 Topic#数, 即为最优主题数目。

该方法可以有效的避免根据经验人为设定主题数目的局限, 提供最优的 LDA 初始主题数目, 从而有效地解决了主题个数的选择问题, 得到更好的模型聚类效果。

[11]

构造主题数如下:

```
def lda_k(x_corpus, x_dict):
    # 初始化平均余弦相似度
    mean_similarity = []
    mean_similarity.append(1)
    # 循环生成主题并计算主题间相似度
    for i in np.arange(2, 11):
        lda = models.LdaModel(x_corpus, num_topics = i, id2word =
x_dict) # LDA 模型训练
        for j in np.arange(i):
            term = lda.show_topics(num_words = 50)
            # 提取各主题词
            top_word = []
            for k in np.arange(i):
                top_word.append([''.join(re.findall('"(.)"', i)) for i in
term[k][1].split(' ')]) # 列出所有词
            # 构造词频向量
```



```

word = sum(top_word, []) # 列出所有的词
unique_word = set(word) # 去除重复的词

# 构造主题词列表，行表示主题号，列表示各主题词
mat = []
for j in np.arange(i):
    top_w = top_word[j]
    mat.append(tuple([top_w.count(k) for k in unique_word]))
p = list(itertools.permutations(list(np.arange(i)), 2))
l = len(p)
top_similarity = [0]
for w in np.arange(l):
    vector1 = mat[p[w][0]]
    vector2 = mat[p[w][1]]
    top_similarity.append(cos(vector1, vector2))
# 计算平均余弦相似度
mean_similarity.append(sum(top_similarity)/l)
return(mean_similarity)

```

### 4.3 主题评价结果

perplexity 是一种信息理论的测量方法，b 的 perplexity 值定义为基于 b 的熵的能量（b 可以是一个概率分布，或者概率模型），通常用于概率模型的比较。

Perplexity 值作为评判标准，评估 LDA 主题模型的好坏。

模型的 perplexity 就是  $\exp\{- (\sum \log(p(w))) / (N) \}$ ， $\sum \log(p(w))$  是对所有单词取 log（直接相乘一般都转化成指数和对数的计算形式），N 的测试集的单词数量（不排重）

```

data.comment_time = pd.to_datetime(data.comment_time)
data['year'] = data.comment_time.dt.year
data['month'] = data.comment_time.dt.month
data['weekday'] = data.comment_time.dt.weekday
data['hour'] = data.comment_time.dt.hour

#各星期的小时评论数分布图
fig1, ax1=plt.subplots(figsize=(14,4))
df=data.groupby(['hour', 'weekday']).count()['cus_id'].unstack()
df.plot(ax=ax1, style='-.')
plt.show()

```



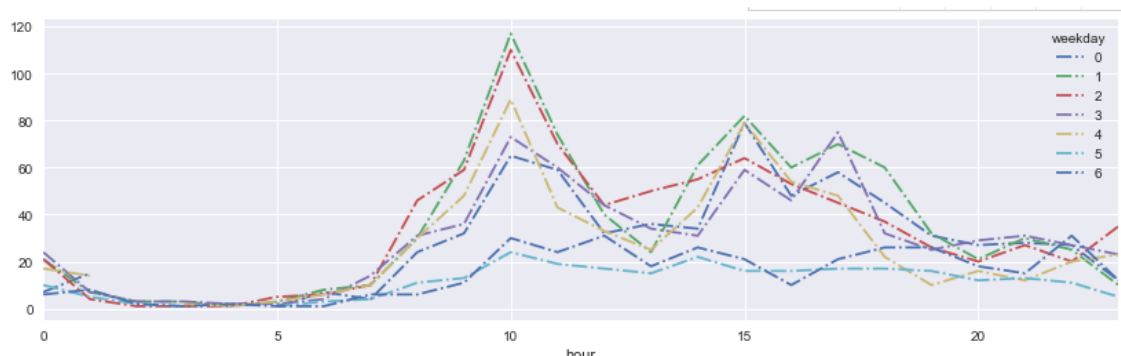


图 16 留言的时序图

根据时序图，我们可以看出，上午 10 点下午 3 点留言人数最多，热点信息较为集中，主题评价结果与时序有关。

## 5. 问题三：答复意见评价

### 5.1 问题分析

针对问题三，需要根据相关部门对留言进行的回复，进行相关性、完整性、以及可行性等角度对答复意见的质量给出一套评价方案。相关性定义:如果回复里出现了和用户有相同的关键字关键字，那说明这条回复是与问题相关的，比如增加快递专柜问题，若相关部门提出，快递专柜或者 A 小区，那说明和留言是有相关性的。完整性定义:如果相关部门针对某问题，提出了这个问题产生的原因或者给出相应的解决办法，那么这个回复就是完整的。可行性定义:回复中，提出的解决方案是有效的，比如，快递问题，如果，回复中有提到说会增加快递专柜，那么就是可行的，如果只是说明了出现这个问题的原因而没有提出具体方案，那这个问题是不可以行的具体的求解步骤如下：

- 1) 根据附件 4 留言和回复的数据，对其进行数据的预处理；
- 2) 处理后的数据根据其相关性、完整性、可行性的比较，分为 A、B、C 三类；
- 3) 分别对这三个类里的数据进行质量的评级，确定为 a、b、c 三个等级；
- 4) 对给出的等级构建质量评价模型

### 5.2 思路分析

通过观察发现，附件 4 中的表中，有留言用户和详情和时间，答复意见和时间，两类数据。先对留言和回复分别进行预处理，分别比较两者之间的相关性、完整性、可行性等三类，然后进行汇总。首先，查看一下附件 4 的数据，利用以下代码提取

数据，其数据总量：2816

```
import xlrd

Workbook=xlrd.open_workbook("text4.xlsx")

Worksheets=workbook.sheet_names()

Print("worksheet is %s"%worksheets)

Worksheet1=workbook.sheet_by_name(u'Sheet1')

num_rows=workbook.sheet1.nrows

For curr_row in range(num.rows):

Row=worksheet1.row_values(curr.row)

Print(curr_row,row)
```

其中部分数据如下表所示

留言详情	答复意见
<p>2019 年 4 月以来，位于 A 市 A2 区桂花坪街道的 A2 区公安分局宿舍区（景蓉华苑）出现了一番乱象，该小区的物业公司美顺物业扬言要退出小区，因为小区水电改造造成物业公司的高昂水电费收取不了（原水电在小区买，水 4.23 一吨，电 0.64 一度）所以要通过征收小区停车费增加收入，小区业委会不知处于何种理由对该物业公司一再挽留，而对业主提出的新应聘的物业公司却以交 20 万保证金，不能提高收费的苛刻条件拒之门外，业委会在未召开全体业主大会的情况下，制定了一高昂收费方案要各业主投票，而投票不采用投票箱只制定表格要物业公司人员这一利害关系机构负责组织，对投票业主隐私权没有任何保护，还对投反对票的业主以领导等工作方式要求改变为同意票，这种投票何来公平公正公开，面对公安干警采用这种方式投票合法性在哪？</p>	<p>现将网友在平台《问政西地省》栏目向胡华衡书记留言反映“A2 区景蓉花苑物业管理有问题”的调查核实情况向该网友答复如下：您好，首先感谢您对我们工作的信任和支持，关于您在平台栏目给胡华衡书记留言，反映“A2 区景蓉花苑物业管理有问题”的情况已收悉。现将我们调查处理情况答复如下：经调查了解，针对来信所反映的“小区停车收费问题”，景蓉华苑业委会于 2019 年 4 月 10 日至 4 月 27 日以“意见收集方式”召开了业主大会，经业委会统计，超过三分之二的业主同意收取停车管理费，在业主大会结束后业委会也对业主提出的意见和建议进行了认真梳理归纳并进行了反馈，业委会制定的停车收费标准不高于周边小区价格。针对来信所反映的“物业公司去留问题”，5 月 5 日下午，辖区桂花坪街道牵头组织社区、物业公司、业主委员会、业主代表的会议，区住房和城乡建设局也参加了会议。在综合各方面的意见后，辖区桂花坪街道、区住房和城乡建设局已要求业委会依法依规召开业主大会，</p>

	<p>根据业主大会的表决结果再执行相应的程序。再次感谢您对我区工作的理解和关心。2019年5月9日</p>
<p>潇楚南路从2018年开始修，到现在都快一年了，路挖得稀烂用围栏围起，一直不怎么动工，有时候今天来台挖机挖两几下，过几天又来挖几下，对当地的交通和店面的生意带来很大影响，里面的车出去和外面的车进来要绕很大一个圈，很不方便，请有关部门对此监管一下，这路修的时间也太长了，至少可以一段一段的修好，方便街上的老百姓出行。</p>	<p>网友“A00023583”：您好！针对您反映A3区潇楚南路洋湖段怎么还没修好的问题，A3区洋湖街道高度重视，立即组织精干力量调查处理，现回复如下：您反映的为潇楚大道西线道路工程项目，该项目位处于坪塘老集镇，目前正在进行土方及排水施工。因该项目为城市次大道，设计标准高，该段原路基土质较差，需整体换填，且换填后还有三趟雨污水管道施工，施工难度较大，周期较长。加之坪塘集镇原有管线、排水渠道较多，需先处理管线和渠道才能进行道路施工，且因近期持续雨天，为保证道路施工质量，需在晴好天气才能正常施工。目前该项目已完成75土方及50排水，预计今年8月底将完工通车。感谢您对我们工作的关心、监督与支持。</p> <p>2019年4月29日</p>
<p>地处省会A市民营幼儿园众多，小孩是祖国的未来，但民营幼儿园教师一直都是超负荷工作且收入又是所有行业最低，甚至连养老和医疗金都没交，在国家大力倡导普惠型幼儿园的同时更是加大了教师的工作压力，在降低成本的同时还增加了学生数量，让本来就喘不过气的教师更是雪上加霜，希望市委市政府加快提高民办幼儿园教师工资待遇水平和降低工作压力有何具体政策和行动？</p>	<p>市民同志：你好！您反映的“请加快提高民营幼儿园教师的待遇”的来信已收悉。现回复如下：为了改善和提高民办幼儿园教师待遇，根据2019年1月8日出台的《中共A市委A市人民政府关于学前教育深化改革规范发展的实施意见》长发〔2019〕2号文件精神，对于学前教育教师的培养和待遇问题做出了明确要求。一是在提高教师待遇方面，依法保障民办幼儿园教职工待遇，民办幼儿园聘任教职工要依法签订劳动合同，依法缴纳城镇企业职工养老保险、医疗保险、生育保险、工伤保险、失业保险和住房公积金，民办园要参照当地公办园教师工资收入水平，合理确定相应教师的工资收入。二是加强监管协同推进，加强对民办幼儿园的日常监管和质量管理，保障民办幼儿园教师待遇，在完善人事（劳动）、工资待遇、社会保障和职称评聘等方面继续推进。感谢您对我市学前教育的关注和支持！</p>
<p>尊敬的书记：您好！我研究生毕业后根据人才新政落户A市，想买套公寓，请问购买公寓能否享受研究生3万元的购房补贴？ 谢谢。</p>	<p>网友“A000110735”：您好！您在平台《问政西地省》上的留言已收悉，市住建局及时将您反映的问题交由市房屋交易管理中心办理。现将相关情况回复如下：按照《A市人才购房及购房补贴实施办法（试行）》第七条规定：新落户并在A市域内工作的全日</p>

	<p>制博士、硕士毕业生（不含机关事业单位在编人员），年龄 35 周岁以下（含），首次购房后，可分别申请 6 万元、3 万元的购房补贴。</p> <p>“首次购房”是指在 A 市限购区域内首次购买商品住房（含住宅类公寓）。因此，如购买商业性质公寓（非商品住房），则不可申领购房补贴。以上情况，望您知晓和理解。如您还有疑问，建议可拨打市房屋交易管理中心咨询电话 0000-00000000 详询。特此回复！2019 年 4 月 30 日</p>
--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 5.3 数据预处理

在本模型中，我们主要是对于某条答复意见。得出这条答复意见与留言的相关性和完整性以及可行性的等级，为了能够建立这种模型，首先，我们采用独热表示（One-Hot Representation）和分布式表示（Distributed Representation）来完成自然语言处理数字化表示。如图 17 所示。

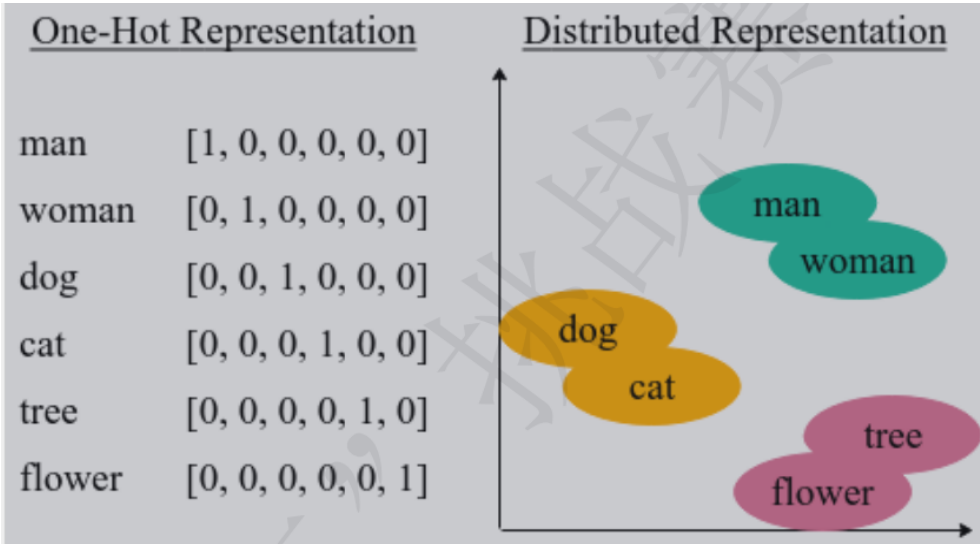


图 17 语言数字化示意图

其中，独热编码是将每个词用 0 和 1 构成的稀疏向量来进行表示，其向量维度是词典大小，所有维度中只有一个元素为 1。然而这种表示方法容易主要存在两个问题，一是容易导致“维度灾难”的发生，当维度增加时，所需存储空间呈指数增长。另一个重要问题就是“词汇鸿沟”，也就是说任意两个词之间都是孤立的，光从这两个向量看不出两个词是否存在关系。

分布式表示是一类将词的语义映射到向量空间中的自然语言处理技术，每一个词用特定的向量来表示，向量之间的距离在一定程度上表征了词之间的语义关系，

即两个词语义相近，在向量空间的位置也相近。

在对数据集分布式表示之前，我们需要对数据集进行预处理，将问题和回答转换为数字化词向量。主要包括分词，字典化、序列化以及填充字符。具体吧步骤如下。即：

步骤 1：去重。将文本中出现相同的内容进行剔除选其一，使得文本内容更可靠。此时利用的代码为：

```
data_dup = data_new['message'].drop_duplicates
```

步骤 2：去除 X 序列操作，将其文本中用 xx 代表的内容去除。利用的代码为：

```
import re  
  
data_qumin=data_dup.apply(lambda x:re.sub('x',"",x))
```

步骤 3:分词。采用 python 自然语言处理工具 jieba, jieba 分词不仅可以使用自己本身的词典，还可以自己指定自定义的词典，这样可以更好的包含 jieba 词库中没有的词。利用的代码为：

```
import jieba  
  
jieba.load_userdict('newdic1.txt')  
  
data_cut=data.qumin.apply(lambda x:jieba.lcut(x))
```

步骤 4：去停用词。将分词后的词语编号，映射到一个数字，以标识词语。

利用的代码为：

```
stopWords=pd.read_csv('stopword.txt', encoding='GB18030',  
,sep='hahaha', header=None)  
  
stopWords=['会','月','日']+list(stopWords.iloc[:0])  
  
data_after_stop=data_cut.apply(lambda x:[i for i in x if i not in  
stopWords])
```

步骤 5：序列化。将一个句子中的词语序列化成词向量列表

```
import json  
  
info = {"留言详情":" 1", "留言内容":" 2",}
```

```
f = open("json 序列化.txt","w",encoding="utf-8")
print(json.dumps(info))
f.write(json.dumps(info))
f.close()
```

步骤 6：填充字符。神经网络的输入数据为固定长度，因此需要对序列进行填充或截断操作。小于固定长度的序列用 0 填充，大于固定长度的序列被截断，以便符合所需的长度

```
str.rjust(width,'0')
input: '798'.rjust(32,'0')
output: '000000000000000000000000000000000000798'
```

步骤 7：封装。将经过以上步骤的文件进行封装以便更好的进行选择。此操作代码为：

```
def data_process(file='text4.xlsx'):
    labels=data_new.loc[data_after_stop .index,'label']
```

经过上述预处理流程，我们构建了训练数据集和测试数据集：语料字典、留言集、答复意见集。这些词向量数据集，将用于后续的构建模型，可以更好的适应自然语言处理任务。相关情况为：

关于H市电商产业园扶持创业的咨  
询H市地区新生儿落地险办理问题  
建议H市国家森林公园安置路牌'  
H市学院在军训期间无理收费',

根本解决不了学生上下学，群众上下班的问题。  
学生学有所长，有利于培养“合格+特长”的学生。

## 5.4 关键词匹配

在这个问题中，我们根据 TF-IDF<sup>[7]</sup>模型对留言与答复的相关意见进行相关性、完整性、可行性的求解，具体情况如下：

首先，把语料库中每个短路表示成向量空间模型。为了方便阐述，做以下符号定义：句子 d 中出现所有词语的集合标记为  $W = (w_1, w_2 \dots w_M)$ 。通过 TFIDF 算法，可以得到留言中的每个词语 TF-IDF 值的向量，记做  $t = (t_1, t_2 \dots t_M)$ ，其中  $t_1$  表示  $w_1$  在留言中的 TF-IDF 值,  $t_2$  表示  $w_2$  在留言中的 TF-IDF 值。

于是可以将要比较的留言 d1 与答复意见 d2 表示为 TF-IDF 值的向量：

$$d1 = (t_{11}, t_{12} \dots t_{1M}) \quad d2 = (t_{21}, t_{22} \dots t_{2M})$$

最后利用余弦定理计算留言与答复意见的相关性：

$$\cos\theta = \frac{\sum_{i=1}^n t_{1i} * t_{2i}}{\sqrt{\sum_{i=1}^n (t_{1i})^2 * \sum_{i=1}^n (t_{2i})^2}}$$

当余弦值越接近 1 时，表明留言 d1 与答复意见 d2 越相关。

其次，在当相关性为零时，确定为完整性和可行性也为零，此时在麻醉相关性一定的基础上，此时相关性定为 x，完整性为 y, 所以完整性为：把 d3 作为答复意见出现的于留言详情不同的关键词，此时的关键词集合为：d3=(t<sub>32</sub>, t<sub>32</sub>, t<sub>33</sub>... t<sub>1M</sub>)，此时满足

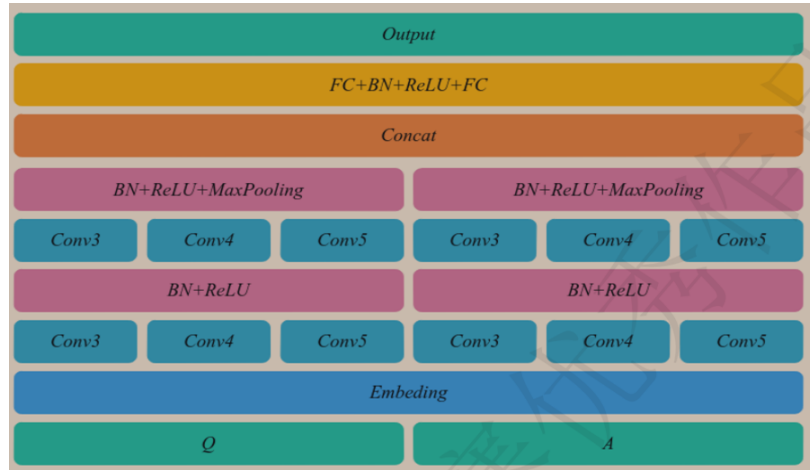
$$y = (\cos\theta) * x + \sum_{i=1}^n t_{i3} * \cos\theta^2$$

此时，确定完整性的情况，当 x 越大的情况下，y 也越大，也即，当相关性越高时，此时出现的解决方案时，证明完整性越高，反之，则越低；最后并且在完整性的基础上，当提出的问题能得到有效实施时，证明其可行性越高，则代表可行性越低。

## 5.5 模型设计

我们提出一个精准匹配模型，该模型拥有多个输入（即成对输入的留言和答复意见）以及单输出（输出 0 到 1 之间的浮点数，其中 0 代表问答毫无关系，1 代表问答完全匹配）

此简化模型为：



```
#评论的长短可以看出评论者的认真程度
data['comment_len'] = data['reviews'].str.len()
fig2, ax2=plt.subplots()
sns.boxplot(x='comment_star',y='comment_len',data=data, ax=ax2)
ax2.set_ylim(0,600)
```

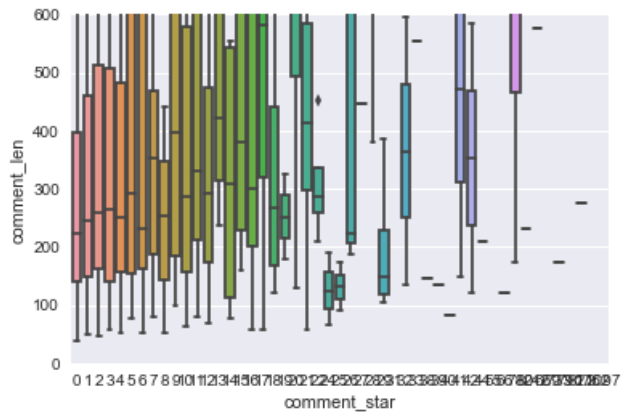


图 18 评论长短与认真程度关系示意图

由此可以看出点赞数越多评论长度更短，看来短一点的评论才更有力度，答复意见更具可信力和认可度，答复意见的评价标准与点赞数和评论长短有关。



## 参考文献

- [1]郑捷. NLP 汉语自然语言处理原理与实战[M]. 北京:电子工业出版社, 2017:94-96.
- [2]闫立坛. 基于机器学习的勘探门户新闻自动分类研究[D]. 西安石油大学, 2019.
- [3]陈丛丛. 主题爬虫搜索策略研究[D]. 山东大学, 2009.
- [4]梁入文. 基于文本意见挖掘的烟台大学教学评价系统设计与实现[D]. 电子科技大学, 2015.
- [5]关满祺. 一种快速的短文本相似度检测方式[J]. 通讯世界, 2020, 27(01):29-30.
- [6]华秀丽. 朱巧明, 李培峰. 语义分析与词频统计相结合的中文文本相似度量方法研究[J]. 计算机应用研究, 2011, 29(3):834-836
- [7]华秀丽. 朱巧明, 李培峰. 语义分析与词频统计相结合的中文文本相似度量方法研究[J]. 计算机应用研究, 2011, 29(3):834-836