基于网络问证平台的数据挖掘与综合分析

摘要

本文旨在基于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的本文信息，通过 python 自然语言处理和文本数据挖掘，对文本信息进行量化处理后分析，进行标签分类处理，热度和留言答复的评判标准，对提升政府的管理水平和施政效率提供有效的帮助。

针对问题一，利用 python 中 基于 sklearn 的朴素贝叶斯分类器将文本转化为 TF-IDF 向量。分成七个类别之后，通过建立混淆矩阵，显示预测标签和实际标签之间的差异。最后借助 F-score 指标来评估模型，得到对分类方法的评价。

针对问题二，第一步建立贝叶斯平均模型和牛顿冷却定律模型探讨时间范围内的高频词的热度处理，然后对两种模型的结果赋权，最后利用时间序列图，详细地区分短期、长期与周期性热点随时间变化。第二步利用贝叶斯平均法对梯度分数进行修正处理点赞数和反对数对高频词热度的影响。对两个步骤的结果进行赋权得到最终的热度排名。

针对问题三，采用基于向量空间模型的计算方法，将文本转化为向量的形式，通过此向量计算文本的相似度，并选择了 Jaccard 相似度模型作为评判的标准，答复的相关性角度对答复意见的质量给出对应评价方案；采取判断留言回复对留言详情中所有问题的解决程

度，进行相关性分析，以判断答复的完整性；通过计算出所有的留言时间与答复时间之差的均值，作为衡量回复时间的标准。最后通过赋权建立对回复意见的质量的评价方案。

关键词：TF-IDF 算法，朴素贝叶斯分类，线性支持向量机模型，牛顿冷却定律模型，Jaccard 相似度模型。

目录

一.问题分析

1.对问题一的分析

对于附件二的数据进行分析，利用多标签分类，根据附件二中的留言详情，利用 python 中文文本分析将文本分成七个类别（城乡建设，环境保护，交通运输，教育文体，劳动和社会保障，商贸旅游，卫生计生）且每条数据只能对应七类中的一类。我们选取附件二中一级分类和留言详情两列数据，首先清洗数据中的空值，然后统计各个类别的数据量。

	一级标签	...	cut_留言详情
0	城乡建设	...	A3 ',' 区 ',' 大道 ',' 西行 ',' 便道 ',' 未管 ',' 路口 ',...
1	城乡建设	...	位于 ',' 书院 ',' 路 ',' 主干道 ',' 在水一方 ',' 大厦 ',' 一楼...
2	城乡建设	...	尊敬 ',' 领导 ',' A1 ',' 区苑 ',' 小区 ',' 位于 ',' A1 '...
3	城乡建设	...	A1 ',' 区 ',' A2 ',' 区华庭 ',' 小区 ',' 高层 ',' 二次 '...
4	城乡建设	...	A1 ',' 区 ',' A2 ',' 区华庭 ',' 小区 ',' 高层 ',' 二次 '...
...	...	...	...
9205	卫生计生	...	夫妻 ',' 农村户口 ',' 女 ',' 岁 ',' 岁 ',' 半才 ',' 15 ',...
9206	卫生计生	...	2015 ',' 年 ',' 月 ',' 16 ',' 号 ',' B ',' 市中心 ',...

图 1：数据量分布图

2. 对问题二的分析

我们将对文本的处理分为两个大的部分。首先探讨时间范围内的高频词的处理，得到初步的综合热度值数据排名。第二部分，处理点赞数和反对数对高频词的处理，得到初步的热词 2 排名。再通过，对决定综合热度值排名的系数和热词 2 排名的系数进行处理，通过分析与统计，定义相应的权重。最终决定热词的排名。

3. 对问题三的分析

通过对答复的相关性，完整性，可解释性，答复与留言的时间差四个方面来建立评价体系。相关性通过计算文本相似度来实现，完整性通过判断留言回复对留言详情中所有问题的解决程度，可解释性通过所给的答复意见是否可以让意见用户清晰明了地理解该问题如何得到解决来衡量，时间差通过计算出所有的留言时间与答复时间之差的均值，作为衡量回复时间的标准。

二．数据预处理

2.1 剔除异常样本

通过对问题的分析，数据样本中有空值栏，我们将这些空值进行删除，得到有效数据。这是对数据的第一步操作。在这个基础上，我们才能对数据进行有效的操作。

2.2 删除与分析无关指标

通过对每个问题方向的研究，根据每道题不同的需要，提取需要的指标，删除无关指标。

三．模型建立

3.1 问题一

数据预处理，我们将七个分类用 0-6 分别表示，并且对中文进行，删除文本中的标点符号，特殊符号，无意义的常用词等一系列的文本

删除工作。经过分词后，在剩余文本 cut-review 基础上生成了每个分类的词云，我们保留各个分类中的前 100 个高频词。

下一步，我们使用的方法抽取文本特征值，用以计算 cut-review 的 TF-IDF 特征值。TF-IDF 是一种用于信息检索与数据挖掘的常用加权技术。目的是，在文本单词的计数前提下，降低高频词的权重，提高罕见值的权重。

为了处理简化核心高频词，提高准确度，我们使用卡方检验的方法找出每个分类中的关联度最大的两个词语和词语对。利用 sklearn 中的 chi2 方法来实现该卡方检验。

为了训练分类器，我们使用了 sklearn 的朴素贝叶斯分类器，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。训练完成后，我们让它预测自定义的 review 的分类。选择几个模型中平均准确率高的模型。

对于该模型，我们通过建立混淆矩阵，显示预测标签和实际标签之间的差异。混淆矩阵的主对角线表示预测正确的数量，其余都是预测错误的数量。之后我们借助 F-score 指标来评估模型。

3.1.1.1 分析数据

对于不同的问题，我们对数据有不同的处理方式。

对于问题一，我们统计了附件二数据条数，统计出一共 条数，接着我们依照表里的一级分类的分类，统计出各个类别里的数据量，比如说，城乡建设类的留言有几条..... 一直将所有的数据归类到

一级标签标明的种类里去。我们此时就只选择出了留言编号，一级分类，留言详情这几栏数据来进行接下来的分析。

在剔除空值之后，我们发现各个类别的数据量不一致，为了更加清晰的了解各种分类下的留言数量，接下来，我们用图形化的方式列出了各个类别的分布

图例如下：

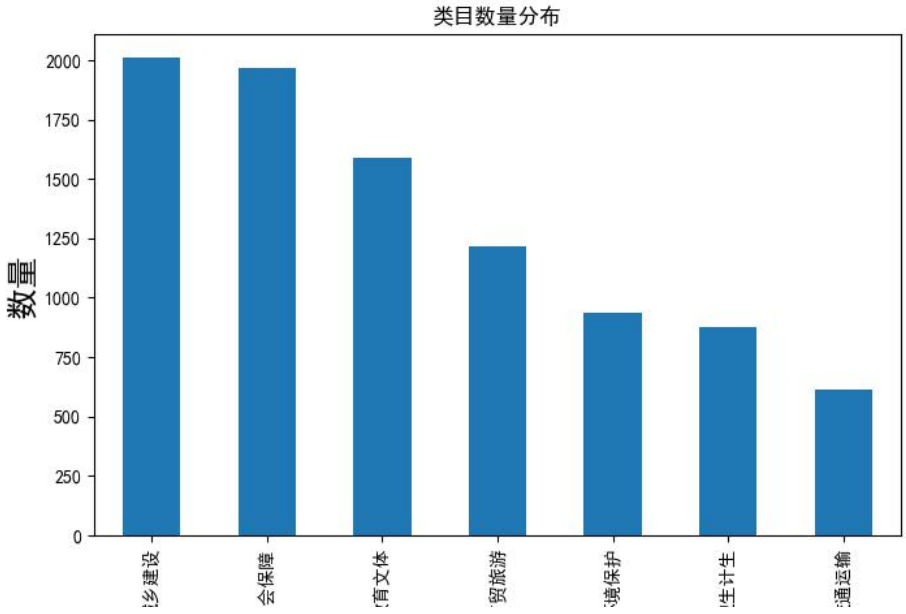


图 2：类目数量分布

接下来，我们将一级标题量化，按照表内的顺序，将七类分别从 0-6 编号，这样处理，便于之后对分类模型的训练。

3.1.1.1 文本处理

本题分析的依据主要是留言详情，所以我们要对留言详情进行中文文本处理。

中文的多样性和复杂性,在留言中有些不必要的标点符号,特殊符号,以及一些无意义的停用词。这些在留言内容中大量存在，但对于我们的分类分析没有任何帮助，反而会增加计算的复杂程度和系统开销，所以我们首先对留言详情的内容进行清洗。

中文停用词包含了许多日常使用频率挺高的常用词,像是一些感叹词，这些不是实词，无法反映出文本的具体意思，所以我们先删去留言的中文停用词。

接下来，我们在删去了停用词的基础上，对留言评价进行分词，把每个评价内容分成由空格隔开的一个一个的单独的词语。

经过分词以后，每个词语中间是由空格隔开然后，我们生成每个分类的词云，我们要在每个分类中罗列前 100 个高频词。

3.1.2 提取特征值

接下来，我们计算留言详情中的 TF-IDF 的特征值，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 意思是词频(Term Frequency)，IDF 意思是逆文本频率指数(Inverse Document Frequency)。TF-IDF 是在单词计数的基础上，降低了常用高频词的权重,增加罕见词的权重。因为罕见词更能表达文章的主题思想,比如在一篇文章中出现了“中国”和“卷积神经网络”两个词,那么后者将更能体现文章的主题思想,而前者是常见的高频词,它不能表达文章的主题思想。所以“卷积

神经网络”的 TF-IDF 值要高于“中国”的 TF-IDF 值。这里我们会使用 slearn 特征抽取方法来抽取文本的 TF-IDF 的特征值。

TF-IDF 该算法用一种统计学的方法来衡量一个词语在文本中的重要程度，常被用于信息提取，文本挖掘中，该算法的核心是计算一个文本中某个词语的 tf 值和 idf 值。

1.TF 词频（term frequency ）衡量一个词语在文档中出现的频率

TF=特征词在文章中的出现次数/文章中的总词数

TF 的定义里包含了标准化的思想，不是直接的使用次数

2.IDF 反文档频率（inverse document frequency ）某一个给定词语在文档集合中出现的次数与文档总数的商

IDF=log（留言总个数/包含当前词的文档个数+1）

从表达式可以看出：当前词所在的留言个数越多，IDF 的值越小，说明该词不重要；反之，该词越重要；IDF 更像是给 TF 赋的一个权重。

3.TF-IDF 用来抽取文章关键词的方法先计算每个词在文章中的词频（TF），计算词的权重（IDF），最后将 TF、IDF 相乘在排序，得到 top N 个关键词，也就是文章的关键词

TF-IDF 计算公式：

TF-IDF=TF×IDF

这里我们使用了参数 ngram_range=(1,2),这表示我们除了抽取留言中的每个词语外，我们还抽取了每个词相邻的词并组成一个“词语对”。这样我们就扩展了我们收集词库的数量，丰富的词库大大提高了我们分类文本的精确度。参数 norm=‘12’，是一种数据标准化处

理的一种方式，可以将数据限制在一定的范围内，例如（-1，1）。我们可以发现维度为（1，2）这里 1 表示我们一共有 1 条留言详情，2 表示我们的特征数量（包括全部留言中所有的词语数以及词语对）。

接下来，我们使用卡方检验的方法来找出每个一级标题分类中关联度最大的两个词语以及词语对。

卡方检验是一种统计学的工作，用来检验数据的拟合度和关联度。所以我们使用 sklearn 中 chi2 方法。

经过卡方检验后，我们列出了每个分类中关联度最强的两个词和两个词语对。这些都可以较好的反映出分类成一级标题的主题。

一级标题分类	交通运输	劳动和社会保障	卫生计生	商贸旅游	城乡建设	教育文化	环境保护
词语	快递	退休	医生	猪肉	小区	文化	环保局
	出租车	职工	医院	广告	业主	学校	污染
词语对	出租车公司	失业保险	乡村医生	检验检测	住房公积金	培训机构	环境监测站
	出租车司机	最低工资标准	人民医院	资质认定	汽车检测站	学校老师	生态破坏

表 1：一级标题主题词汇

3.1.3 分类器的选择

我们首先将留言转变为包含数字的词向量。分完词向量之后，我们开始训练我们的分类器。

分类是数据挖掘的一种非常重要的方法。分类的概念是在已有数据的基础上学会一个分类函数或构造出一个分类模型（即我们通常所说的分类器(Classifier)）。该函数或模型能够把数据库中的数据纪录映射到给定类别中的某一个，从而可以应用于数据预测。总之，分类器是数据挖掘中对样本进行分类的方法的统称，包含决策树、逻辑回归、朴素贝叶斯、神经网络等算法。

3.1.3.1 朴素贝叶斯分类器

朴素贝叶斯分类是一种十分简单的分类算法，叫它朴素贝叶斯分类是因为这种方法的思想真的很朴素。朴素贝叶斯的思想基础是这样的：对于给出的待分类项，求解在此项出现的条件下各个类别出现的概率，哪个最大，就认为此待分类项属于哪个类别。

3.1.4 模型的选择与评估

通过如上的分析，我们尝试了不同的及其学习模型，并评估了他们的准确率，我们将使用如下四种模型作为测试，选择精确度最好的作为本题的分类模型

1.Logistic Regression(逻辑回归)

2.(Multinomial) Naive Bayes(多项式朴素贝叶斯)

3.Linear Support Vector Machine(线性支持向量机)

4.Random Forest(随机森林)

我们画出每种模型的箱体图，我们可以得出哪种准确率最高或者最低。一般来说集成分类器的准确率很低，不适合处理高维度数据，因为文本中有大量的数据有很多的特征值，使得集成分类器难以应付，其中最高的是线形向量机的准确率最高。如图所示

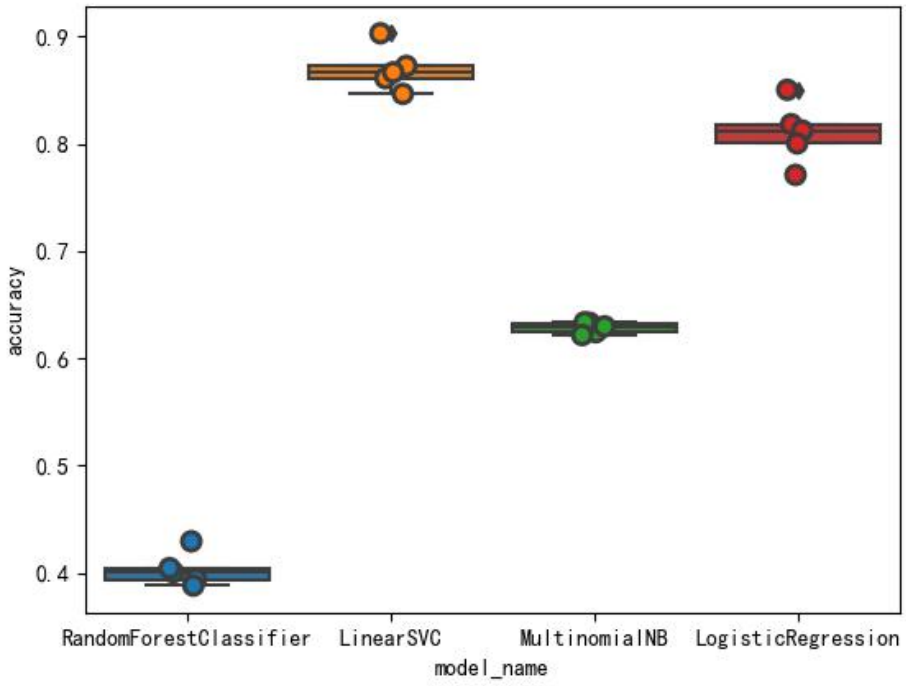


图 3：模型准确率预测

根据我们上述选择出的准确度最高的模型，我们继续查看混淆矩阵并显示预测和实际标签之间的差异。

混淆矩阵的每一列代表了预测类别，每一列的总数表示预测为该类别的数据的数目；每一行代表了数据的真实归属类别，每一行的数据总数表示该类别的数据实例的数目。每一列中的数值表示真实数据

被预测为该类的数目。混淆矩阵的主对角线表示预测的正确的数量，除主对角线以外的其余都是预测错误的数量。

多分类模型一般不使用准确率来评估模型的质量，因为准确率不能反映出每一个分类的准确性，因为当训练数据不平衡时，准确率不能反映出模型的实际预测精度，这时，我们就需要借助 F-sorce 方法。

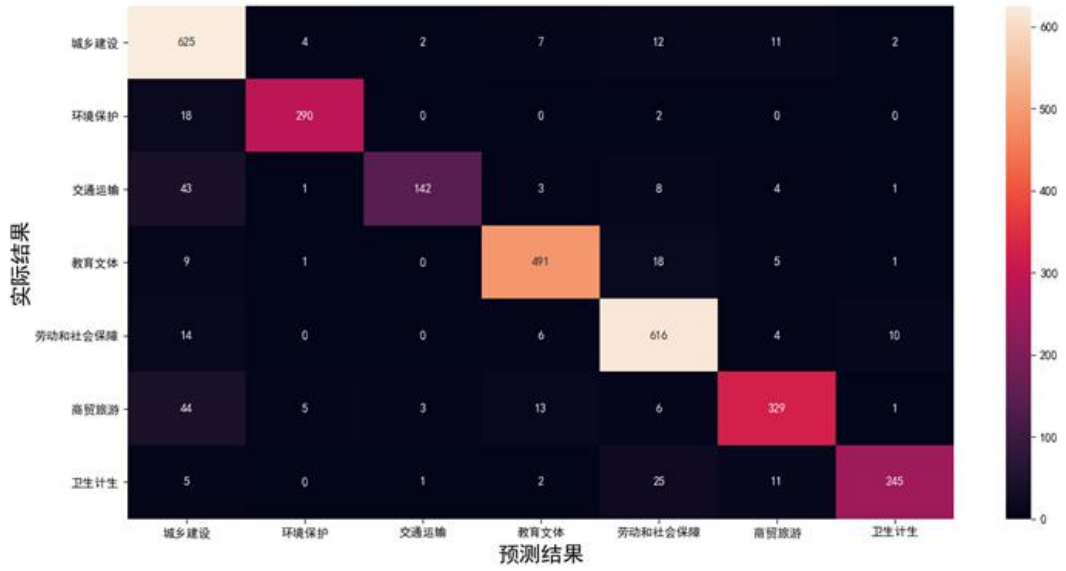


图 4：混淆矩阵

下面，我们将查看各个类的 F1 分数，如下图所示

$$F1 = \frac{1}{n}\sum_{i=1}^{2}\frac{2P_iR_i}{P_i + R_i}$$

	precision	recall	f1-score	support
城乡建设	0.82	0.94	0.88	663
环境保护	0.96	0.94	0.95	310
交通运输	0.96	0.7	0.81	202
教育文体	0.94	0.94	0.94	525
劳动和社会保障	0.9	0.95	0.92	650
商贸旅游	0.9	0.82	0.86	401
卫生计生	0.94	0.85	0.89	289

从上图的F_1分数上看，交通运输和教育文体的F_1分数最大为 0.76 而最小的F_1分数为商贸旅游，这一类的准确率较高，但召回率较低，所以F_1分数才较小。

3.2、问题二

3.2.1 时间处理

以一周为时间单位，研究每周的词频热度，研究某一段特定时间内的词频热度。

3.2.2 模型简介

我们通过使用贝叶斯平均法和牛顿冷却定律法来分别定义并计算每一周为单位的词频的热度 1。通过对两种方法的结合，对两种方法求出的热度 1 设置权值，得到最终的综合热度值。

$$WR = \frac{v}{v+m}R + \frac{m}{v+m}C$$

贝叶斯平均法：

WR 是每个词的加权得分，WR 越大表示热度越大

R 是该词汇的平均得分（这里设定都为 1），v 是总词频

m 是排名前 n 的词汇的最低词频（n 是自定义的阀值）

牛顿冷却定律

将热词排名比喻成一个即自然冷却的过程。可以利用物理学定律，建立“温度”与“时间”之间的函数关系，构建一个“指数式衰减”的过程。　　牛顿冷却定律：物体的冷却速度，与其当前温度与室温之间的温差成正比。

定义负冷却系数：

$$\partial(w)=\log_e\left(\frac{\text{当前词频}+1}{\text{历史词频}+1}\right)/\text{时间差}$$

负冷却系数越高则说明热度就越大。

3.2.3 热词的评价标准

（1）我们选择计词语一周内的词频，词频在当天未峰值，并大于某一阀值

（2）该峰值与起始值差值大于某一阀值

（3）热度值大于某一阀值 H(w)为热度

Atp(w)当前词频　　all(w)表示以 Ayp(w)为中心的前后三天的词频　　BeTp(w)为起始词频　　c1,c2 为阀值。

求解步骤：

（1）通过 python 进行分析留言详情选取处理后的前 200 高频词（不包括无意义的词）。我们通过文本片段出现的频数、文本内部聚合度、粘联度三方面判断一个文本片段是否能够独立成词。

文本片段的出现频数：

如果一个文本片段在语料中多次出现,那么它有可能是一个词,反之,只是偶然出现的字词组合很难认定为独立的词。 我们的目的是检测热点话题,出现频率很少的词不太可能为实时热点,可以忽略,同时也可以快速排除大量候选词,加快算法速度。因此本文规定一个文本片段的出现频数应超过某个阈值,否则不作为候选词。

3.2.4 文本片段的内部聚合度

原理：构成词的字之间必然存在一定相关性,而不仅仅是几个字的随机组合。

假设长度为 n 的文本片段 X 由字 $x_1x_2x_3\cdots x_n$组成,Count(X)表示 X 在训练语料中出现的次数。 我们将文本片段 X 看作字符串 X_1与 X_2 的组合,则 P(X) =P(X_1)P(X_2 | X_1)。 对于长度为 n 的文本片段 X 有 n -1 种可能的分割方式。 根据最大似然估计的估算公式，P(X_2 | X_1) 即 Count(X_1X_2) / Count(X_1)。 根据已有研究结论,用互信息度量字串内部紧密性的效果最佳。

已知文本片段 X 看作文本片段 X1 与 X2 的组合,这个事件的互信息为后验概率与先验概率比值的对数:

$$MI = \log \frac{p(x)}{p(X_1)p(X_2)}$$

为减少计算量,仅取上式中的真数部分作为字串内部聚合度的度量。 按 X_1、X_2 所有组合分别计算出这个比值,取其中的最小结果作为文本片段 X 的内部聚合度。 之后对其设定阈值,达不到阈值的文本片段不作为候选词。 在实际计算时,由于客观条件限制,无法使用大规模训练语料来估计参数,因此使用 X 在样本语料中出现的次数代替 X 在训练语料中出现的次数。实验发现取此近似值不会对抽词效果产生重大影响,但可以极大地简化算法。 如果 X 为二字词,简化后的公式为:

$$H(x) = -\sum_{i=1}^{n} p(x_i)\log_b p(x_i)$$

其中,P(X) 表示文本片段 X 的内部聚合度,Length 表示整个样本语料的长度,Count(X) 表示 X 在训练语料中出现的次数。文本片段的粘联度

$$P(X) = \frac{\text{Count}(X)}{\text{Count}(X_1)\,\text{Count}(X_2)} \times \text{Length}$$

本文使用信息熵来量化文本片段的粘联度。 在信息论中,熵被用来衡量一个随机变量出现的期望值。 信息熵的计算公式为:

其中 p(x_i) 表示事件 x_i 发生的概率，{x_i⋯ , x_i} 为 x 的集合。 b 是对数所使用的底数,通常取 2、10 或自然常数 e。 这里选择 e 作为底数。信息熵直观反应一个离散事件有多随机,随机性越大信息熵就越大。

（2）采用归一法来处理词频数据（今天的词频/（昨天的词频+今天的词频））

（3）利用贝叶斯平均法，计算每一周内所有词的热度 1

（4）提到的牛顿负冷却定律的公式对同样的高频词进行计算，可以求出每个词的负冷却系数。负冷却系数越高则说明热度越大。

（5）接下来对两组计算出的值各自设置一个权值，得到综合的 H（w）热度值。

公式如下：

$$H(w)=\alpha * B(w)+\beta * \partial(w)$$
$$Atp(w)>c1,$$
$$B(w)>c2,$$
$$\partial(w)>c3,$$
$$Atp(w)-Btp(w)>c4.$$

Atp(w)为当前词频 B(w)为贝叶斯平均值 α (w)为负冷却系数

Btp(w)为历史词频

（6）最后利用时间序列图，可以更详细地区分短期、长期与周期性热点画出所有的高频词随时间变化的热度 1 曲线。进行比较分析对高频词的综合的 H（w）热度值数据排序。

3.2.4.1 点赞数，反对数数据处理

（1）数据处理

首先利用上文 python 处理过后的高频词。

（2）进行热度分数计算：

利用贝叶斯平均法对梯度分数进行修正处理。例如：对于用户的投票排名，用户投票评分的人很少，则算平均分很可能会出现不够客观的情况。这时引入外部信息，假设还有一部分人（C 人）投了票，并且都给了平均分（m 分）。把这些人的评分加入到已有用户的评分中，再进行求平均，可以对平均分进行修正，以在某种程度或角度上增加最终分数的客观性。容易得到，当投票人数少的时候，分数会趋向于平均分;投票人数越多，贝叶斯平均的结果就越接近真实投票的算术平均，加入的参数对最终排名的影响就越小。

时间和点赞数综合热度排名：

通过数据分析，确定与时间相关的综合的 H（w）热度值和点赞数反对数得到的热度 2 的权值。从而得到最终的热度排名。

3.2.4.2 问题描述的话题提取

我们运用上面得到的最终的热度排名的前 50 个词语表，通过提取所有含有该 50 个词的数据列，然后再利用 python 进行处理，我们使用了 TF-IDF 算法对文档进行了向量化，把每条由词语组成的句子转换成一个数值型向量，把所有的数据转换为词频矩阵作为 Kmeans 模型的输入，TF-IDF 最大特征值选择为 20000。

词频矩阵形成后，我们直接调用 sklearn 的 Kmeans 模型，对所有数据进行聚类。

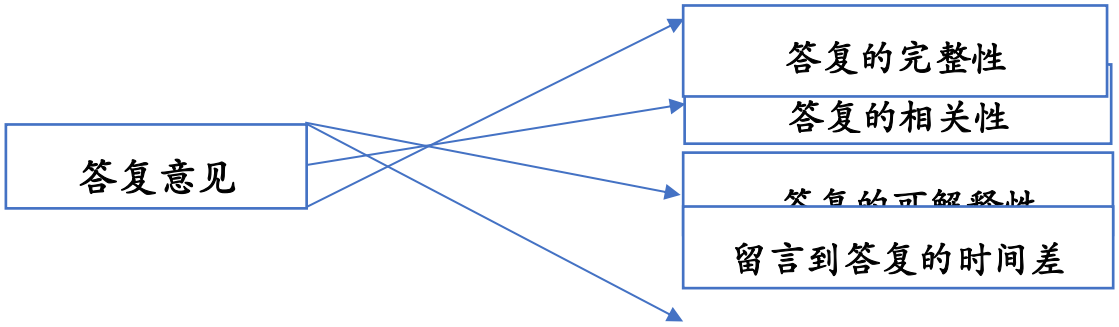
随后，我们使用 matplotlib 来绘制聚类结果，并将每类的前五条数据的信息输出。

最后通过聚类信息的关键词，通过概况连接成一个简单易理解的问题描述。整理得到最后的热点问题的 excel 表为：

热度排名	问题 ID	热度指数	时间范围	地点/人群	问题描述
1	1	5	2019/6/15 至 2020/1/15	A2 区丽发新城	搅拌站粉尘噪音污染
2	2	4	2019/5/23 至 2019/9/25	A5 区劳动东路魅力之城小区	夜宵摊污染问题
3	3	3	2019/2/15 至 2019/7/16	A7 县幼儿园	普惠幼儿园问题
4	4	2	2017/6/8 至 2019/11/27	西地省经济学院	强制学生实习问题
5	5	1	2019/1/10 至 2019/11/12	A 市	交通规划建设问题

3.3、问题三

由题目可知我们的方面分析的结构如下：



3.3.1 答复的相关性分析

我们采取对附件 4 中留言详情与留言的答复意见进行相关性分析。采用基于向量空间模型的计算方法，将文本转化为向量的形式，通过此向量计算文本的相似度。经过对各种模型的对比，我们最终选择了 Jaccard 相似度模型（杰卡德相似度）作为评判的标准，答复的相关性角度对答复意见的质量给出对应评价方案。

3.3.1.1 模型简介

向量空间模型，在此模型中，文本被看作是由一系列相互独立的词语组成的，若文档 D 中包含词t_1，t_2，…，t_N。则文档表示为 D（t_1，t_2，…，t_N）。由于文档中词语对文档的重要程度不同，并且词语的重要程度对文本相似度的计算有很大的影响，因而可对文档中的每个词语赋以一个权值 w，以表示该词的权重，其表示如下：D（t_1，w_1；t_2，w_2；…；t_n，w_N），可简记为 D（w_1，w_2，…，w_N），此时的w_k即为词语 t_k的权重，1≤k≤N。这样，就把文本表示成了向量的形式，同时两文本的相似度问题也就可以通过两向量之间的夹角大小来计算了，夹角越大，两文本的相似度就越低。

杰卡德相似度，指的是文本 A 与文本 B 中交集的字数除以并集的字数，杰卡德相似度与文本的位置、顺序均无关。计算 Jaccard 距离公式为

3.3.1.2 求解步骤

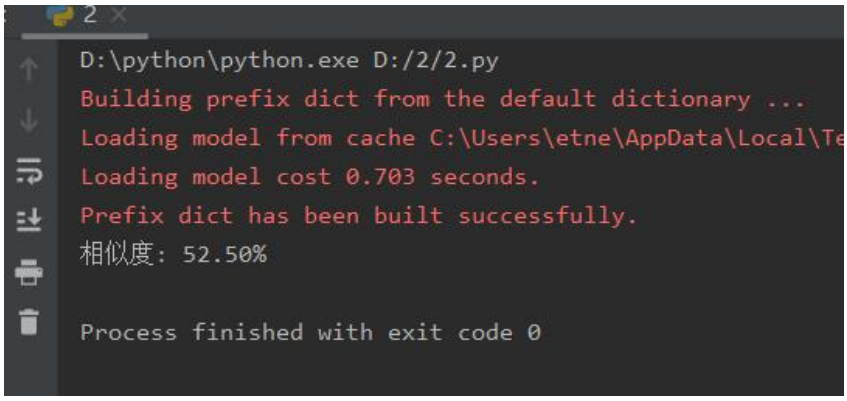
1. 用 python 删除空值等基本数据处理后，并且将“转给相关部分”等类似留言信息单独提出，不进行相关性统计，然后对剩余数据进行提出词语频率的排名。

2. 提取词语作为文本向量的特征词。

在文本中出现频率较高的词语应该具有较高的权值，因此，在计算词语对文本的权重时，应考虑词语在文本中的出现频率，记为 tf。去除一些不具备鉴别能力的高频词语。因而，在计算词语权重时还应考虑词语的文本频率（df）,即含有该词的文本数量。由于词语的权重与文本频率成反比，又引出与文本频率成反比关系的倒置文本频率（idf），其计算公式为 idf=logN/n（其中 N 为文本集中全部的数量，n 为包含某词语的文本数）。由此得出特征词 t 在文本 D 中的权重 weight(t,D) = tf(t,D) * idf(t)。用 tf*idf 公式计算特征项的权重,既注重了词语在文本中的重要性,又注重了词的鉴别能力。因此,有较高的 tf*idf 值的词在文本中一定是重要的, 同时它一定在其它文本中出现很少。因此我们通过这种方法来选择把那些词语作为文本向量的特征词。

3.特征词选择出来之后我们就确定了文本的向量，我们通过此向量运用余弦相似度模型来计算文本的相似度。余弦度越接近于 1，则说明留言详情和答复意见的相似度越高。

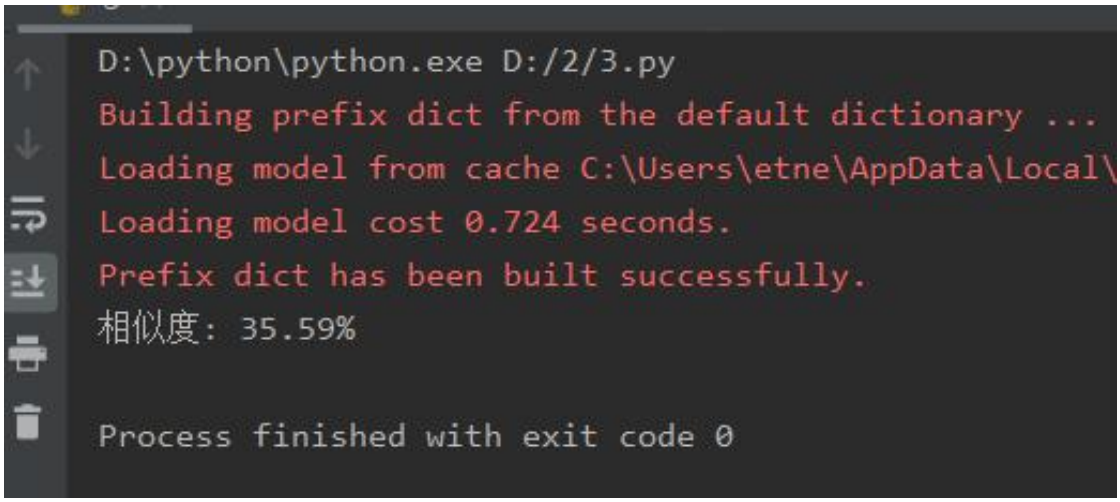
4.结果如下：



3.3.2 答复的完整性分析

求解步骤：

首先分别对留言详情和回复的文本内容正则过滤 html 标签，然后将 html 转义符实体化，对文本内容进行切割。之后提取关键词，去除停用词，然后进行分词和关键词一起提取，进行 Jaccard 的相似度计算。最后一步进行除零处理，测试结果如下：



模型简介：

Jaccard 模型:

Jaccard 相似系数又称为 Jaccard 相似性度量（Jaccard 系数，Jaccard 指数，Jaccard index）。用于比较有限样本集之间的相似

性与差异性。Jaccard 系数值越大，样本相似度越高。定义为相交的大小除以样本集合的大小：

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}=\frac{|A\cap B|}{|A|+|B|-|A\cap B|}$$

（若 A B 均为空，那么定义 J（A，B）= 1）

Jaccard 距离（Jaccard distance），定义为 1- Jaccard 系数，即：

$$d_j(A,B)=1-J(A,B)=\frac{|A\cup B|-|A\cap B|}{|A\cup B|}=\frac{A\Delta B}{|A\cup B|}$$

3.3.3 答复的可解释性分析

可解释性，我们认为是所给的答复意见是否可以让意见用户清晰明了地理解该问题如何得到解决，即在答复意见中是否存在过多的无效回答或冗余词汇。所以求解步骤如下：先利用 jieba 分词将答复意见进行分词，根据有关资料建立一个无关词库（包含与留言主题相关性较小的词语），再利用 python 将已分好词汇中的无关词去掉，得到有关词组。之后再计算出有关词组占分出的所有词组的比例，得到有效比例，再根据规定的评价标准，进而判断答复的可解释性。

3.3.4 评价方案结果

综合上面三种因素，我们可以得出每一条答复的好坏程度,我们分别将这三种因素平均决定最后答复的好坏程度，权重为 10/3:10/3:10/3。

四．参考文献

[1] 顾摇 森．基于大规模语料的新词发现算法[J]．程序员,2012(7):54-57

[2] 钟摇 将,耿升华,董高峰．一种新词检测方法研究[J]．数字通信,2013,40(2):1-5.

[3]基于朴素贝叶斯的文本分类研究综述[J]．贺鸣,孙建军,成颖．　情报科学．2016(07)

[4]基于 K-means 算法的神经网络文本分类算法研究[J]．卢曼丽．　中国管理信息化．2014(21)