总体介绍：

论文中用 re 模块、sklearn 库、pandas 库、 jieba 库、numpy 库等 python 工具对数据进行处理，在建模中用到了多项式朴素贝叶斯 MultinomialNB 模型和 LdaModdel 模型.代码在 jupyter notebook 中用 python 语言实现.

一、 群众留言分类

需要按照附件 1 中给出的三级标签体系，给附件 2 中的留言(如下图)做一级标签分类.

留言编号	留言用户	留言主题	留言时间	留言详情	一级分类
744	A089211	建议增加 A 小区快递柜	2019/10/18 14:44	我们是 A 小区居民...	交通运输

留言主题是留言详情的概括，所以选择留言主题的数据进行处理以及模型构建.关键步骤如下：

- 文本预处理

1. 利用 re 模块去掉留言主题中无用的符号：\t，\n，\r 等等

这里说的无用是指对留言主题的语义以及留言的分类无用. 比如\t，\n，\r 这些字符不影响留言主题的语义，还会对分类结果造成干扰，需要使用 re 模块去掉.

2. 利用 jieba 库对留言主题做分词处理

留言主题中涉及到很多地名，比如 A 市、I3 县、C1 区、荆城小区、经济学院等等. 在主题中把这些地名提取出来(具体见问题二的解题思路） ，以自身字典的形式导到 jieba 中，保证地名的完整性.

3. 去停用词

在分词后，有些词语对于留言主题的语义理解是冗余的，比如你，我，您好，了，的以及一些标点符号等等. 这里采用了泰迪云课堂上《基于文本内容的垃圾短信分类》的学习资料中的 stopword、百度停用词表、哈工大停用词表、中文停用词表、四川大学机器智能实验室停用词库的合并版对留言主题做去停用词处理，然后根据去词效果再增添 stopword.

- 建立分类模型

1. 数据划分

将全部数据用sklearn.model_selection.train_test_split划分成9/10为训练集，

1/10 为测试集. (经测试，相对 test_size=0.05，0.15，0.2，0.25，当 test_size=0.1 时分类得到的 f1_score 最高.)

2. 文本向量化

计算机只能处理结构化数据，而留言主题的内容是文本，属于非结构化数据，所以需要把数据抽象成数学符号表示.

借助 CountVectorizer 将文本转换成文本词条矩阵，再借助 TfidfTransformer 计算 TF-IDF 值，给较关键的词以较大权重.

3. 多项式朴素贝叶斯分类模型(相对高斯朴素贝叶斯，分类效果更好)

将数据导入 MultinomialNB()中，利用 classification_report 得出分类报告，利用 confusion_matrix 得出混淆矩阵.

● 结果及分析

1. f1_score

	precision	recall	f1-score	support
交通运输	0.55	1.00	0.71	31
劳动和社会保障	0.94	0.73	0.82	259
卫生计生	0.69	0.98	0.81	61
商贸旅游	0.64	0.93	0.76	89
城乡建设	0.91	0.71	0.80	265
教育文体	0.81	0.83	0.82	142
环境保护	0.76	0.97	0.85	74
accuracy			0.80	921
macro avg	0.76	0.88	0.80	921
weighted avg	0.84	0.80	0.81	921

一级标签共有 7 类，由上图可看出 f1_score 的 macro avg 即均值是 0.80. 从 f1_score 来看，该分类模型对环境保护类留言的分类效果最好（0.85），对交通运输类留言的分类效果最差（0.71）.

对于交通运输类、卫生计生类、商贸旅游类、教育文体类、环境保护类这 5 类留言的分类，查全率(recall)比查准率(precision)高；对于劳动和社会保障类、城乡建设类这 2 类留言的分类，查全率(recall)比查准率(precision)低.

2. 混淆矩阵

```
[[ 31   0   0   0   0   0   0]
 [  9 189  18  13  11  14   5]
 [  0   0  60   1   0   0   0]
 [  3   0   1  83   1   0   1]
 [ 11   6   5  27 188  13  15]
 [  2   7   2   5   6 118   2]
 [  0   0   1   0   1   0  72]]
```

该矩阵与上面的 classification_report 对应，行和为该类留言的真实数量，列和为预测为该类留言的数量. 比如，在测试集中，有 5 条劳动与社会保障类留言、1 条商贸旅游类留言、15 条城乡建设类留言、2 条教育文体类留言被误分到环境保护类，而实际为环境保护类的 74 条留言中有 1 条被误分到卫生计生类中，有 1 条被误分到城乡建设类中

- 改进

一级分类	二级分类	三级分类
城乡建设	安全生产	事故处理
城乡建设	安全生产	安全生产管理
城乡建设	安全生产	安全隐患

附件 1 中给出的是三级标签分类体系(如右图)，但上面的分类模型中并没有考虑到留言的二级分类、三级分类. 如果将二级分类以及三级分类的标签利用起来，即当留言主题中出现二三级分类中比如安全管理、安全隐患这样的词时，直接将该条留言贴上“城乡建设“的一级标签，这样分类效果应该会有所改善.

.

二、 热点问题挖掘

- 文本预处理

与问题一中步骤大同小异，对附件 3 中留言主题文本处理

- 主题模型

1. 生成语料库

经过文本预处理，每条留言主题都变成由几个词组成，这些词以空格相隔，如下图所示.

```
一米阳光 婚纱 艺术摄影 合法 纳税
道路 命名 规划 初步 成果 公示 城乡 门牌
春华镇 金鼎村 水泥路 自来水 到户
```

将所有留言主题的词语都放到一个 gensim.corpora 的 Dictionary 中，对词语进行编号，并得出每条留言主题对应的数学符号(如下图).

```
[(0, 1), (1, 1), (2, 1), (3, 1), (4, 1)],
[(5, 1), (6, 1), (7, 1), (8, 1), (9, 1), (10, 1), (11, 1), (12, 1)],
[(13, 1), (14, 1), (15, 1), (16, 1), (17, 1)],
```

解释：（x， y）中，x 表示该词语的编号，y 表示该词语在本留言主题中出现的次数.

2. LdaModel 模型

主题模型算法可以有效地从文本语义中提取主题信息，是对文字隐含主题进行建模的方法. 常用的主题模型有潜在语义索引(LSI)、概率隐含语义标(PLSA)、隐式狄利克雷分布(LDA). 这里采用 IDA 主题模型.

注：在刚使用 LdaModel 模型时，得到的 topic 里有很多指向不明确的词，比如村、站、小区、街道、路、苑、景园等等. 这些词语给探索主题造成阻碍，所以除了将更多的词加入到停用词表 stopword 里，还利用 re.findall 把名称中带有村、站、小区、街道、路、苑、景园等字眼的词（一般表示的是某个地方）提取出来（提取结果见作品附件/地名/），再把这些词复制到自建字典 newdic 中，重新对文本做分词处理.

- 结果及分析

```
0.010*"违规" + 0.008*"诈骗" + 0.007*"公司" + 0.007*"经济学院" + 0.006*"房屋" + 0.006*"家园" + 0.005*"物业" +
0.005*"安全隐患" + 0.005*"学生" + 0.004*"新村社区"
0.014*"扰民" + 0.009*"幼儿园" + 0.008*"油烟" + 0.006*"项目" + 0.006*"渣土" + 0.006*"时代" + 0.005*"规划" + 0.
005*"魅力之城" + 0.004*"公园" + 0.004*"市场"
0.010*"影响" + 0.007*"居民" + 0.007*"劳动东路" + 0.006*"房屋" + 0.006*"购房" + 0.005*"生活" + 0.005*"补贴" +
0.005*"商铺" + 0.005*"施工" + 0.004*"社保"
0.010*"车位" + 0.009*"业主" + 0.008*"销售" + 0.008*"滨河苑" + 0.008*"伊景园" + 0.008*"捆绑" + 0.007*"规划" +
0.006*"开发商" + 0.006*"公交车" + 0.005*"开"
0.019*"扰民" + 0.015*"噪音" + 0.008*"业主" + 0.008*"地铁" + 0.007*"国际" + 0.007*"经营" + 0.005*"污染" + 0.00
5*"魅力之城" + 0.005*"施工" + 0.005*"夜宵"
```

上图是由 LdaModel 模型探索出的 5 个 topic. 再根据其中的词语查找对应的留言信息，然后统计留言时间，得出热点问题表. xls . 同样，根据词语定位到对应的索引，将同一 topic 的留言排到一块，得到热点问题留言明细表. xls .

- 改进

留言编号	留言用户	留言主题	留言时间	留言详情	反对数	点赞数
188006	A000102948	阳光婚纱艺术摄影是否合法经	2019/2/28 11:25:05	，因为地处居民楼内	0	0
188007	A00074795	洛命名规划初步成果公示和城	2019/2/14 20:00:00	0年都未曾更换过，	0	1

附件 3 中的每条留言信息最右侧都有对该条留言的点赞数与反对数(见上图)，上面的 LdaModel 模型在探索隐含主题的时候没有把这条信息考虑进去，只是单纯根据词频来找热点问题. 如果考虑点赞数和反对数，以点赞数-反对数+1 表示该条留言的热度，点赞数-反对数+1 得到的值越高代表该条留言的热度越高，这样探索出的热点问题应该更加准确.

三、 答复意见的评价

答复意见即工作人员对留言做出的回应，里面应包含对留言中所提到的问题的解决态度

和方法等. 根据答复意见是否对留言做出全面回应、所作回应与留言内容是否相关、是否能让人读懂三方面对答复意见做出评价. 留言详情里包含的内容太冗长，所以通过探索留言主题与答复意见的关系来评价答复意见.

- 文本预处理

步骤与问题 1 中大同小异，一点不一样的是不断完善自建的字典 newdic.txt 和停用字表 stopwod.txt .

- 文本向量化

与问题 1 中同

- 答复意见的相关性

将文本向量化之后，每个留言主题和答复意见都抽象成了数学向量，且长度相同，可通过计算每条留言主题与对应的答复意见的余弦相似度来判断答复意见的相关性. 余弦相似度的取值范围为[-1,1]，余弦相似度越大代表答复意见与留言相关性越大.

比如，设留言向量$(x_1,x_2,\cdots,x_n)$，答复向量$(y_1,y_2,\cdots,y_n)$，则两向量的余弦相似度表示为

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

描述的是两向量所成夹角的余弦值.

- 答复意见的完整性

可通过看留言中出现的关键词是否出现在答复意见中，可根据答复意见中所包含的留言的关键词的数量所占该留言词数的百分比来判断答复意见的完整性. 百分比越大，代表答复意见越完整.

- 答复意见的可解释性

四、 参考文献

[1] https://github.com/goto456/stopwords

[2]https://blog.csdn.net/zz_dd_yy/article/details/51926305?utm_medium=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-1&depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromBaidu-1

[3] https://edu.tipdm.org/classroom/119/courses