

# 分论点识别

## 一、任务背景

### 1.1 背景

参考答案:

标题

共享、共建、共担 创新社区治理

“三事分流”工作法，是由部门负责解决“大事”，即管辖事项及公共服务；村（居）委会为主导，社区组织等共同协商解决“小事”，即村（居）公共事项及公益服务；由群众寻求市场服务或自行解决“私事”，即个人事务和市场服务。

开头

这是社区治理的创新，是政府治理和社会调节、居民自治良性互动的方式，是在社区这个“小世界”中，打造一个最优的时代，实现“大事”能快办，“小事”能共办，“私事”能自办，共享共建共担。

过渡段

社区的“大事”要快办，需落实责任，联动各方，提升效率。“民生无小事，枝叶总关情”，只要是涉及群众切身利益的公共事项及公共服务，都是“大事”，是最基本、最迫切的民生需求。“大事”的解决不是某个干部、某个部门的一家之责，需要协调各部门，加强联动，以此精简环节、提高效率。如北京实施的“街乡吹哨、部门报到”，实现群众需求与部门职能的精准对接。群众有所呼，政府有所应，切实提升群众生活的幸福感和获得感。

分论点

社区的“小事”要共办，需调动社区群众参与积极性，共同协商。过去，政府为群众办实事办好事，却大包大揽，基层干部疲于奔命，群众产生“等靠要”的依赖思想。发展为了人民，更要依靠人民，社区治理也要坚持群众路线，听民意、汇民智、集民力。如“人居环境整治义务督察员”及时发现、反馈问题，共建美好环境。既有政府的投入与权力赋予，也有群众的责任共担，才能求得自上而下精细管理与自下而上广泛参与的“最大公约数”。

社区的“私事”要自办，既要发挥群众的自主性，也要注重教育引导。我们常说“清官难断家务事”，“外界力量”有时并不能弄清“私事”的全貌。与此同时，“私事”也涉及到隐私，若处理不当，把握不好“度”，反而可能弄巧成拙，激化矛盾。因此，对于家庭私事，倡导“自办”。但这并不意味着政府、社区的袖手旁观、隔岸观火，需要为其提供必要的心理、家庭、教育、法律等普惠性服务，注重教育和引导。

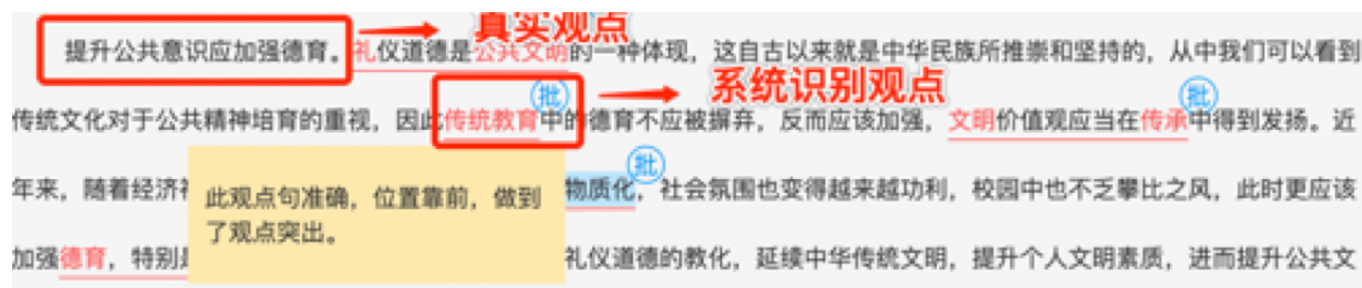
3个论述段

总之，要让“三事分流”工作法真正落地、见实效，需要系统性、全面性、全方位的过程把控。要深入群众，准确把握群众所思所想所盼；要精准识别，合理分类，明确权责；要全面反馈评估，信息公开，接受监督，真正做到准确“找事”、合理“分事”、高效“办事”、全面“反馈评估”。

社区治理没有终点，只有连续不断的新起点，“治大国如烹小鲜”，要继续提升社区治理水平，真正惠民生、暖民心，让社区成为“人人出力、人人尽责”、共享共建共担的大家庭。

结尾

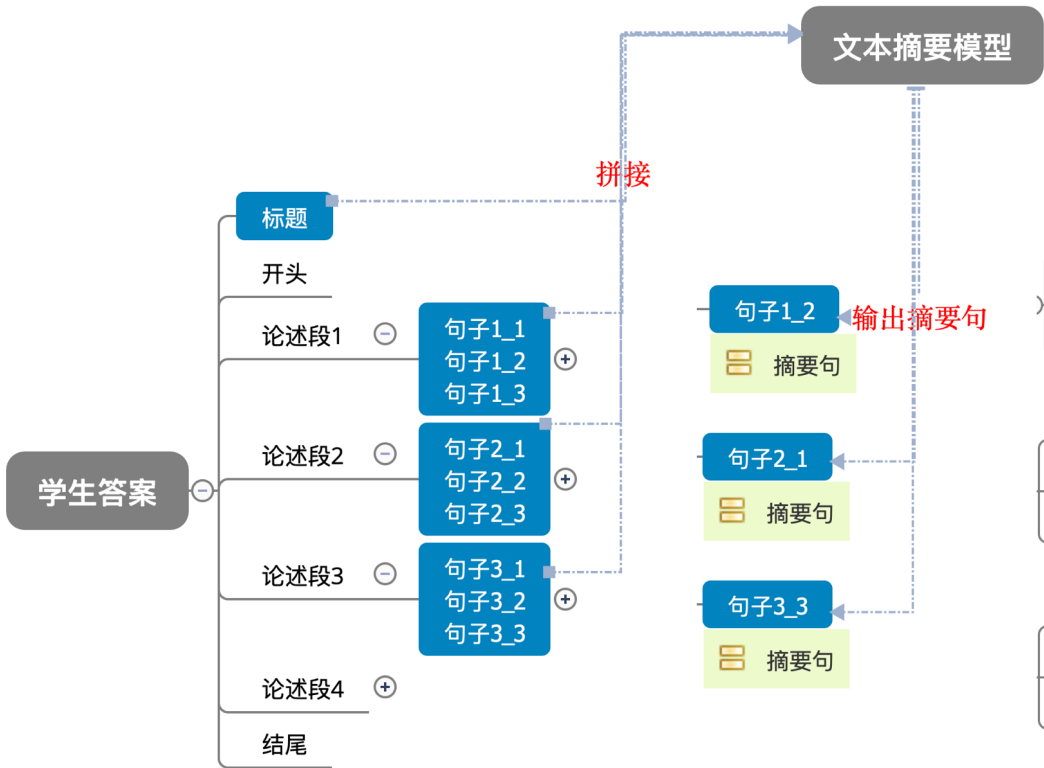
文章结构图



论述段段落图

## 1.2 解决办法

1. 利用文本摘要的方法抽取前80字的关键句，即获得前80字中每句话被选为摘要的概率，取最高的概率即为分论点
2. 由于每个论述段的分论点是相关联的，且和题目也是相关联的，可以作为特征补充第1点。



## 二、数据集构建

### 2.1 标注数据集

每3段构建一个样例，如果一篇文章中有4个论述段

- 4个论述段，123段落为1个样例，234段落为1个样例，一共2个样例。总共1w个样例

### 2.2 伪标注数据集

#### 2.2.1 标注数据集的缺陷

1. 92%的分论点的集中在首句，如下图，使得模型倾向于识别第1句，首次1w数据集实验F1值0.98，见下表
2. 数据集规模较小，泛化性不够

**统计内容**：26798个论述段，每个论述段只取最前一个分论点

**统计指标**：论述段前k句的，准确率P(Precision) 召回率R ( Recall )，，F (前2者的调和平均值)

例子：句子A。句子B(分论点)。句子C。句子D  
计算某论述段前3句的P,R,F,  
R=1. (分论点出现在了前3句)  
P=长度(B)/长度(A+B+C)  
F值就是R,P的平均值

```
dic = {dict} <class 'dict'>: {1: [0.92, 0.77, 0.84], 2: [0.44, 0.91, 0.59], 3: [0.29, 0.95, 0.44], 4: [0.22, 0.97, 0.36], 5: [0.19, 0.98, 0.32], 6: [0.17, 0.99, 0.29], 7: [0.17, 0.99, 0.29], 8: [0.16, 0.99, 0.28], 9: [0.16, 1.0, 0.28], 10: [0.16, 1.0, 0.28]}
1 (4550874752) = {list} <class 'list'>: [0.92, 0.77, 0.84]
2 (4550874784) = {list} <class 'list'>: [0.44, 0.91, 0.59]
3 (4550874816) = {list} <class 'list'>: [0.29, 0.95, 0.44]
4 (4550874848) = {list} <class 'list'>: [0.22, 0.97, 0.36]
5 (4550874880) = {list} <class 'list'>: [0.19, 0.98, 0.32]
6 (4550874912) = {list} <class 'list'>: [0.17, 0.99, 0.29]
7 (4550874944) = {list} <class 'list'>: [0.17, 0.99, 0.29]
8 (4550874976) = {list} <class 'list'>: [0.16, 0.99, 0.28]
9 (4550875008) = {list} <class 'list'>: [0.16, 1.0, 0.28]
10 (4550875040) = {list} <class 'list'>: [0.16, 1.0, 0.28]
__len__ = {int} 10
```

图1：多人标注中取最靠后的分论点

```
前半部分
dic = {dict} <class 'dict'>: {1: [0.8, 0.92, 0.86], 2: [0.35, 0.98, 0.52], 3: [0.22, 0.99, 0.36], 4: [0.17, 0.99, 0.29], 5: [0.14, 1.0, 0.25], 6: [0.13, 1.0, 0.23], 7: [0.12, 1.0, 0.21], 8: [0.12, 1.0, 0.21], 9: [0.12, 1.0, 0.21], 10: [0.12, 1.0, 0.21]}
1 (4457916032) = {list} <class 'list'>: [0.8, 0.92, 0.86]
2 (4457916064) = {list} <class 'list'>: [0.35, 0.98, 0.52]
3 (4457916096) = {list} <class 'list'>: [0.22, 0.99, 0.36]
4 (4457916128) = {list} <class 'list'>: [0.17, 0.99, 0.29]
5 (4457916160) = {list} <class 'list'>: [0.14, 1.0, 0.25]
6 (4457916192) = {list} <class 'list'>: [0.13, 1.0, 0.23]
7 (4457916224) = {list} <class 'list'>: [0.12, 1.0, 0.21]
8 (4457916256) = {list} <class 'list'>: [0.12, 1.0, 0.21]
9 (4457916288) = {list} <class 'list'>: [0.12, 1.0, 0.21]
10 (4457916320) = {list} <class 'list'>: [0.12, 1.0, 0.21]
__len__ = {int} 10
```

图2：多人标注中取最靠前的分论点

**bert初次运行结果：**

	F1	P	R
1w数据集	0.9878283	0.987512	0.9881448
不在首句测试集，26个样例	0.02		

## 2.2.2 扩大数据集至20w

①针对位置的缺陷。移动分论点句子在top80的位置，构建伪标注训练集

例如前80字，有k句话，语序不变情况下，将分论点插入这k个位置。k=3时情况如下

ABC, BAC, BCA

②针对数据集太小。将分论点和该段落非前80字的句子组合成样例

比如段落为：前80字：ABC. 80-200：DEF ----->构建为 ADE

## 2.3 样例构建

### 2.3.1 训练样本

段1前80 + 段2前80 + 段3前80 + 标题

- [CLS]科学，是敢于批判和不断质疑，寻求真相[CLS]以往的科学认知"科学是经验的产物，被证明或重复验证的理论就是正确的[CLS]"但实际中，经积累的产物也极有可能是错误的，如盛行几千[SEP]

- 2 [CLS]想象力需要在已有形象甚至毫无概念的情况下，创造出新的形象[CLS]而艺术将为这一新的形象提供美的元素[CLS]艺术，推出人类无止境地无限美丽的世界前行，为想象力产物注入美[CLS]—[SEP]
- 3 [CLS]古文化，蕴含着无限的智慧和秘密，需要用想象力去充分挖掘[CLS]古文化是想象力保持永不枯竭的动力和源泉，如果不谈古而只论今，抛弃古人遗留下来的智慧，就会丢失一笔宝贵的想[SEP]
- 4 [CLS]"想象力的源泉[SEP]

### 2.3.2 训练标签

```

1 masked_lm_ids:
2 1 0 0
3 0 0 1 0
4 1 0
5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

## 三、BERT-SUM改进

方案：与BERT-SUM不同之处

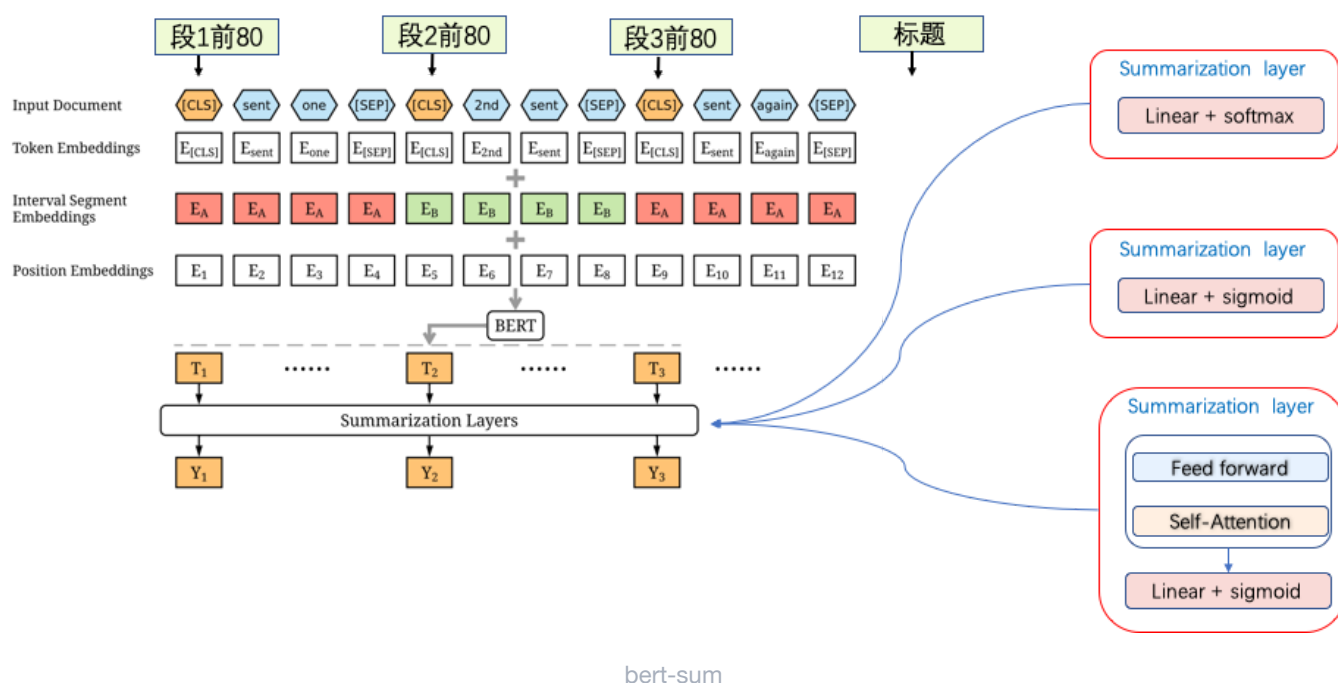
1. 更改loss, CLS输出为摘要句的概率,每个段落取概率top1的为摘要句
2. 输入文本添加了【标题】，起到一个attention的作用

loss为每个cls输出交叉熵，totalloss=Sum(loss[i]), $0 < i < 24$ (每个样例最多24个句子)

模型：linear+softmax：

模型：linear+sigmoid:F1值0.85

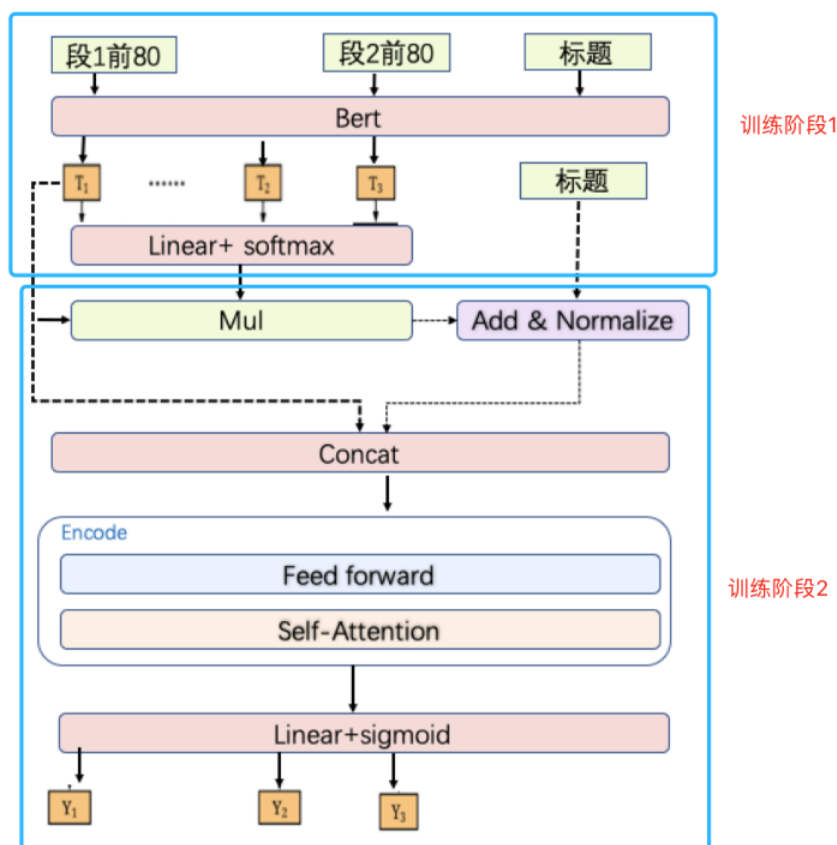
模型：transformer（捕捉句子间关系）+linear+sigmoid



- 「句子编码层」通过BERT模型获取文档中每个句子的句向量编码，

- 「摘要判断层」通过三种不同的结构进行选择判断，为每个句子进行打分，最终选取最优的top-n个句子作为文档摘要

## 四、BERT-SUM + memory + 迁移学习



训练阶段1：「句子编码层」通过BERT模型获取文档中每个句子的句向量编码 [9选3]：选出3个概率最大的句子做损失函数

训练阶段2：

1、bert模型加载训练阶段1的模型参数，并且冻结

2、memory + transformer: 【[9选3]选出的3个句子+标题】，再次作为transformer的输入 结果：F1值 0.89

## 五、业务评估

### 5.1、线上命中规则：

对于每个段落，得分句子的命中规则是分论点，就标为1

e.g.

某个段落，按照句子打标签，比如有5个句子,第4个句子的得分来自于分论点规则命中，则

score=[0,2,0,3,0] -> rule\_label:[0,1,0,1,0]

### 5.2、摘要句模型+规则

1、如果段落有分论点规则命中，只保留模型预测所在句子的标签，其他归零

2、如果段落无分论点规则无命中，不做任何处理

e.g.1: rule\_label:[0,1,0,1,0] ,predict:[0,1,0,0,0] ----> [0,1,0,0,0]

e.g.2: rule\_label:[0,0,0,0,0] ,predict:[0,1,0,0,0] ---->[0,0,0,0,0]

e.g.3: rule\_label:[0,1,0,0,0] ,predict:[0,0,1,0,0] ---->[0,0,0,0,0]

### 5.3、结果

	P	R	F1
规则	0.43	0.19	0.26
模型+规则	0.80	0.18	0.29