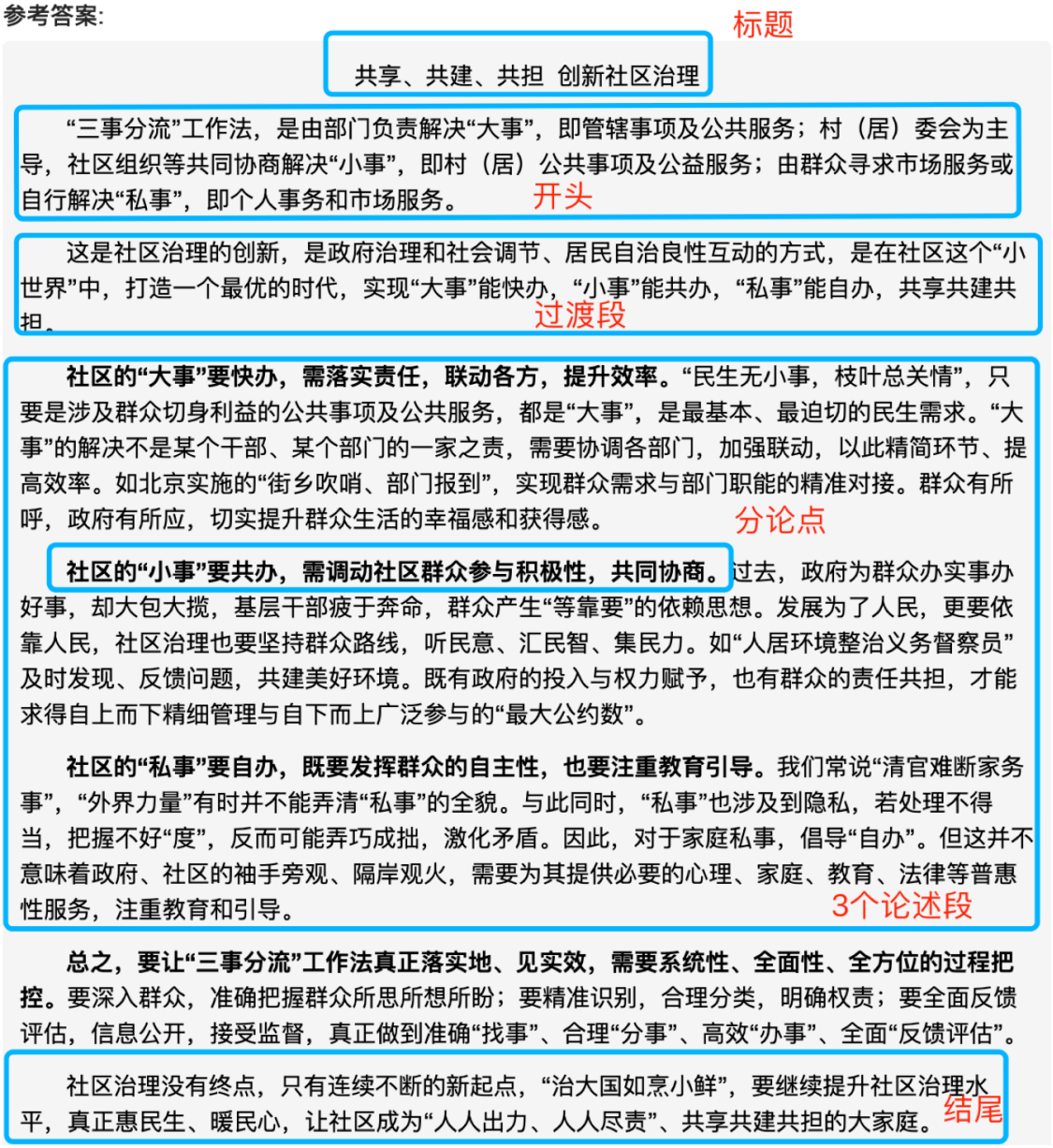
分论点识别

零、目录

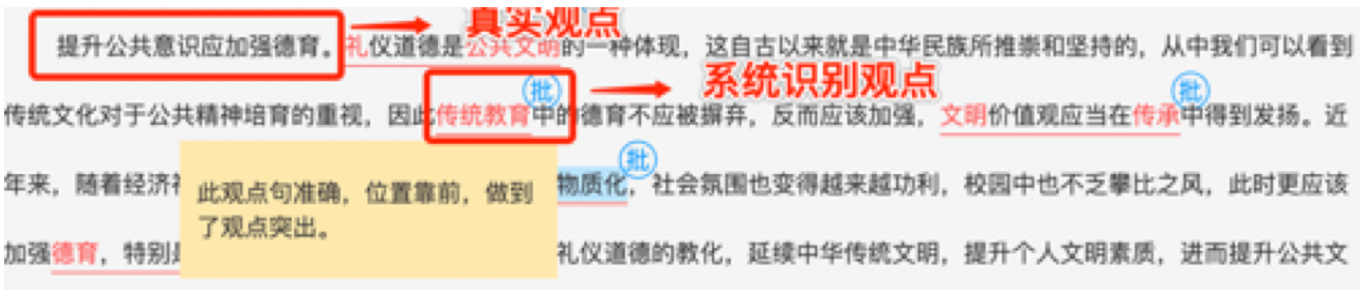
一、任务背景

1.1 背景

参考答案:



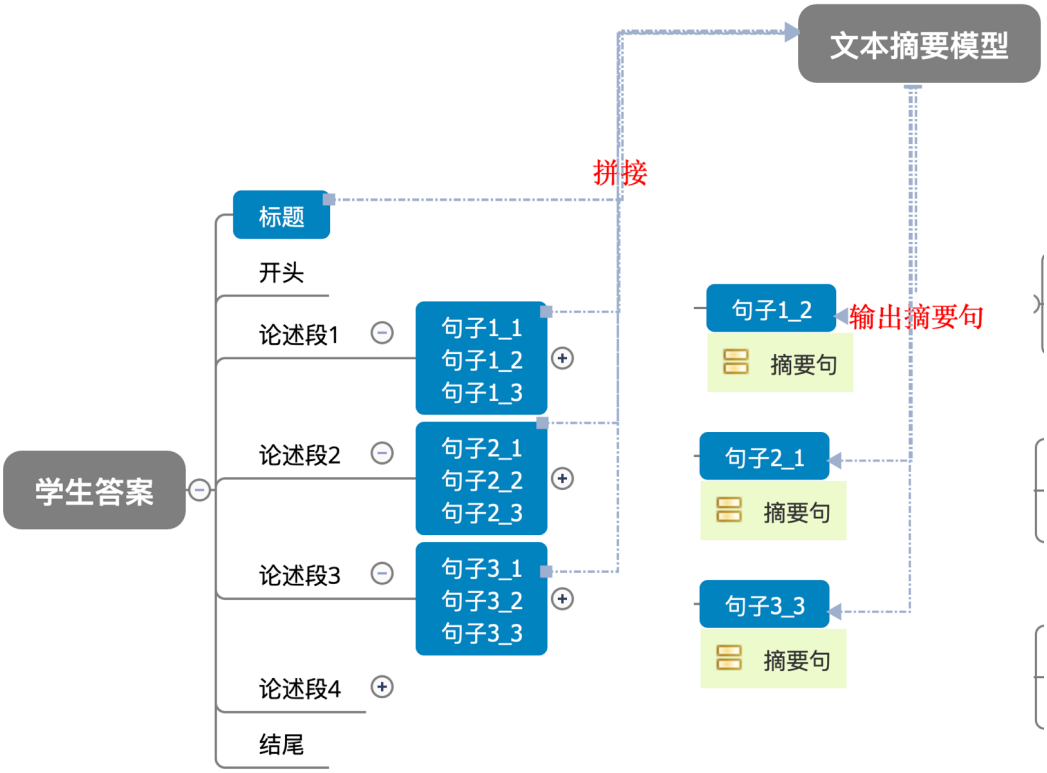
文章结构图



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

图1：多人标注中取最靠后的分论点　　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]
```

```
[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，推出人类无止境地向无限美丽的世界前行，为想象力产物注入美[CLS]一[SEP]
[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]
[CLS]"想象力的源泉[SEP]
```

2.3.2 训练标签

```
masked_lm_ids:
1 0 0
0 0 1 0
1 0
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

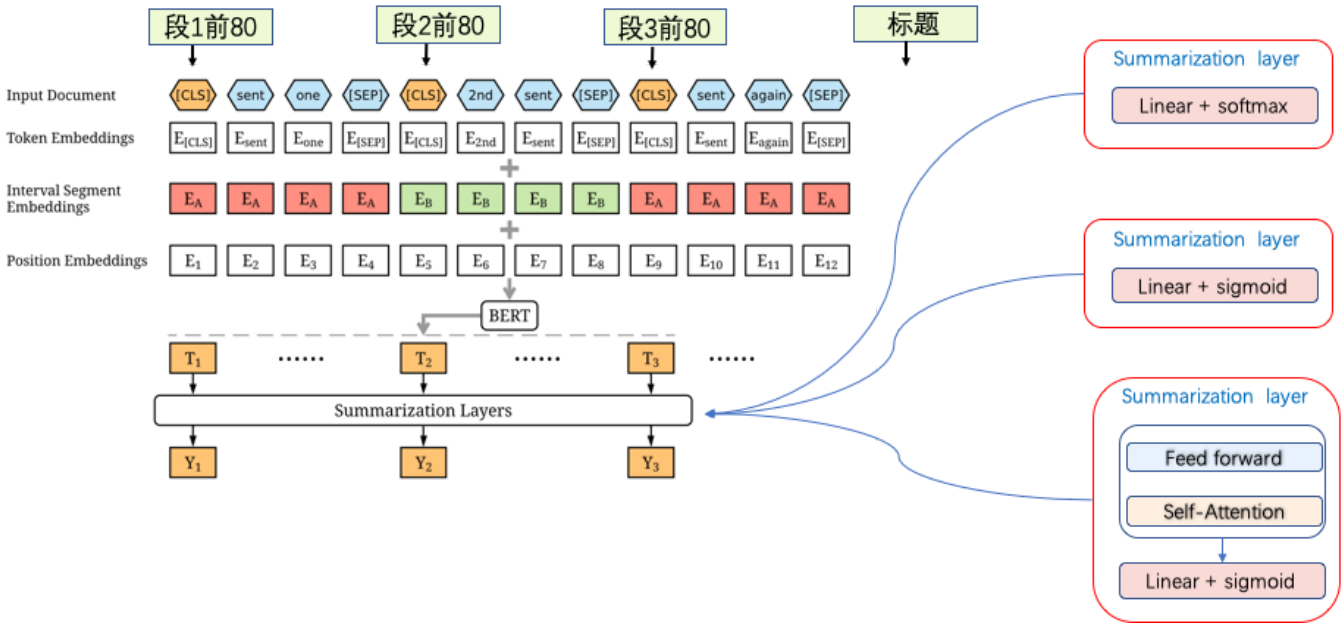
1. 更改loss, CLS输出为摘要句的概率,每个段落取概率top1的为摘要句
2. 输入文本添加了【标题】，起到一个attention的作用

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

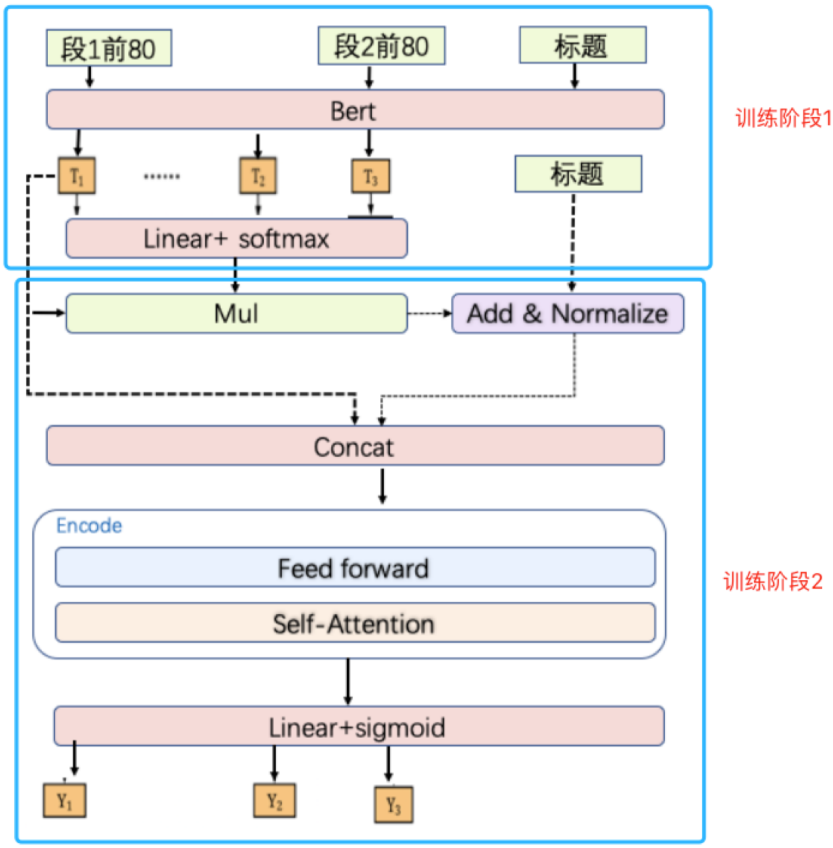


bert-sum

- 「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