DOI:10.19651/j.cnki.emt.2209237

基于 HRAGS 模型的混合式摘要生成方法*

岳 琳 杨凤暴 王肖霞
(中北大学信息与通信工程学院 太原 030051)

摘 要: 针对传统的抽取式、生成式方法在摘要自动生成任务上存在可读性、准确性不足的问题,提出了基于 HRAGS 模型的混合式摘要生成方法。该方法首先使用 BERT 预训练语言模型获取上下文句子表示,结合冗余感知方法构造抽取模型;然后将训练完毕的 BERT 双编码器和随机初始化的具有双编码-解码注意力模块的 Transformer 解码器相结合构造生成模型,采用二阶段微调策略解决编、解码器训练不平衡的问题;最后使用 Oracle 贪婪算法选择关键句作为指导信号,将原文和指导信号分别输入生成模型以获取摘要。在 LCSTS 数据集上进行验证,实验结果表明,相比于其他基准模型,HRAGS 模型能够生成更具可读性、准确性和 ROUGE 得分更高的摘要。

关键词: 预训练语言模型;混合式摘要;生成式摘要;冗余感知
中图分类号: TP391.1 **文献标识码:** A **国家标准学科分类代码:** 520.2

Hybrid summary generation method based on HRAGS model

Yue Lin Yang Fengbao Wang Xiaoxia
(School of Information and Communication Engineering, North University of China, Taiyuan 030051, China)

Abstract: Traditional extractive and abstractive methods lack readability and accuracy in the summary auto-generated task, so a HRAGS model-based hybrid summary generation method was proposed. First, the method used the BERT pre-trained language model to obtain a contextual representation and combined with redundancy-aware method to construct an extractive model. Then a couple of trained BERT encoders were united with a randomly-initialized Transformer decoder contained two encoder-decoder attention modules to construct an abstractive model. The abstractive model adopted a two-staged fine-tuning approach to resolve the training imbalance problem between encoders and decoders. Finally, an Oracle greedy algorithm chose key sentences as external guidance and source document with guidance were put into the abstractive model to acquire a summary, which was verified on the LCSTS evaluation dataset. Experimental results shows that the HRAGS model can generate a more readable, accurate and high ROUGE score summary compared with other benchmark models.

Keywords: pretrained language model;hybrid summarization;abstractive summarization;redundancy-aware

0 引 言

大数据时代的到来使得具有较强聚合和传播功能的移动互联网络平台层出不穷,文本摘要的自动生成作为信息压缩提取的关键技术,不仅能有效解决信息过载的问题,而且可以保障用户利用碎片化时间获取优质实时资讯的需求。因此,如何提升自动文本摘要的可读性和准确性一直是大众研究的热点。

文本摘要的自动生成主要包括抽取式方法和生成式方法。抽取式方法抽取原文中的关键句组成摘要,在语法准确、语义正确等方面具有更好的性能;生成式方法在理解原文的基础上重新组织语言生成摘要,其摘要具有高可读性和连贯性。目前,研究人员深入研究了自动文本摘要领域,提出了许多新颖的模型框架。Jia 等[1]使用深度差分放大器框架来增强摘要句的特征,解决抽取式摘要中句子分类不平衡的问题;Kwon 等[2]提出了一种基于嵌套树结构的抽取式模型来构造信息丰富的摘要;Liu 等[3]提出了生成式摘要框架 SIMCLS,通过对比学习来弥补学习目标与评价指标之间的差距;Wu 等[4]提出了基于统一语义图的生成式摘要框架 BASS 对文本中的显式和隐式关系进行编码,使用图传播注意机制将显式内容选择到摘要中。

然而,抽取式摘要不够精简且句间不连贯[5];生成式摘

收稿日期:2022-03-11
* 基金项目:国家自然科学基金(61972363)项目资助

要容易出现事实错误，其原因是生成过程缺少关键信息的指导[6]。为了解决上述问题，混合式摘要生成方法异军突起。该方法将抽取式与生成式模型相结合，有效提升了生成摘要的可读性与准确性。如 Mendes 等[7]和 Bae 等[8]首先使用抽取式模型从原文中抽取关键句子，然后使用生成式模型对所选句子进行重写和压缩；Krishna 等[9]提出 Cluster2Sent 算法，抽取关键句并聚合为簇，将每个簇生成一个摘要句。考虑到抽取式模型难以获取同时具备多个理想属性的摘要句，Song 等[10]首先抽取出几组不同的候选摘要，然后对其评分、选择和重写；Moroshko 等[11]提出编辑网络思想，在生成阶段对摘要句进行三种后处理操作，即保持原状、重新措辞或直接废弃。为了进一步充实抽取摘要包含的关键信息，Gao 等[12]通过分析原文与摘要之间的相关性来学习摘要模式和原型事实，使用编辑生成器根据抽取的事实生成新的摘要；Shah 等[13]首先从原文中抽取实体对和关系，然后通过聚合操作符建立其比较关系并将其输入生成模型；Zhu 等[14]提出基于主题的混合式模型，首先使用主题检测器来预测输入段落的主题，然后使用指针生成器网络依据主题感知表示来生成摘要。由于内容选择与润色重写相独立的特点，目前混合式摘要生成方法在医疗、司法等不同的场景中发挥着重要作用[15,16]，但该方法仍存在以下局限：1）由于生成阶段建立在抽取阶段的基础上，因此抽取摘要效果的好坏直接影响最终摘要的质量，模型的累计误差较大。2）抽取与生成两阶段间的参数化模型相对独立，模型之间没有参数共享，生成式模型无法充分利用抽取式模型编码的知识，如文献[17]。

为解决上述问题，本文提出了基于 HRAGS 模型的混合式摘要生成方法，提供了抽取式和生成式模型的融合统一体。引入二阶段微调策略[18]，使用训练完毕的抽取式模型[19]参数对生成式模型的编码器进行初始化；生成式模型使用双编码器-单解码器结构[20]分别对原文和指导信号进行编码，具有双编码-解码注意力模块的 Transformer 解码器有效融入指导信号的特征信息，提高了模型的关键信息聚焦能力，减少了对抽取阶段的强依赖关系。在 LCSTS 数据集[21]上的实验结果表明，与基准模型相比，本文模型在 ROUGE 得分[22]和人工评价结果上均有提升。

1　HRAGS 模型构建

本文提出基于 HRAGS 模型的混合式摘要方法，包括摘要抽取模型和摘要生成模型两部分。模型的整体框架结构如图 1 所示。

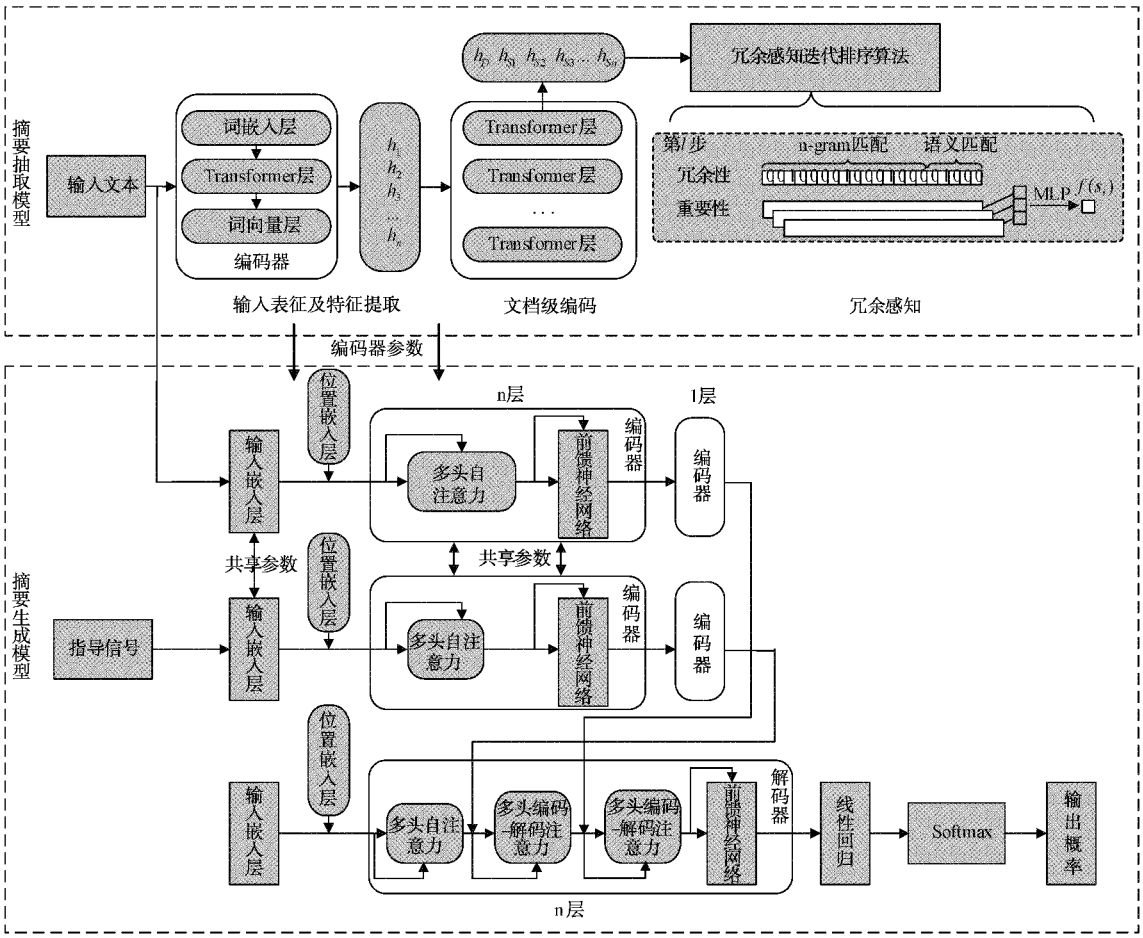

图 1　HRAGS 混合式摘要生成模型整体框架结构

1.1　摘要抽取模型

该模型主要包括输入表征、特征提取、文档级编码和冗余感知[19]等四部分。

1)输入表征

BERT 预训练模型在每个句子首尾添加标识符[CLS]和[SEP]表征句子的开始和结束。对原文进行输入表征,即进行词嵌入、区间段嵌入和位置嵌入操作。词嵌入刻画并融合文本的单字和全局语义信息;区间段嵌入将不同句子以 E_A 和 E_B 区分,并将其映射为向量;位置嵌入对输入的字分别附加一个不同的向量,以区分每个字的位置。

2)特征提取

BERT 预训练模型采用多层双向 Transformer 编码器对文本向量进行特征提取。Transformer 编码器使用融合注意力机制的模型框架,通过缩放点积注意力与多头注意力获取双向语义信息。文本向量 $\boldsymbol{T}=(t_1,t_2,\cdots t_n)$ 经过词嵌入编码操作和多层双向 Transformer 编码器的训练得到文本序列的特征向量 $\boldsymbol{H}^l=(h_1^l,h_2^l,\cdots h_n^l)$。其中 l 是 Transformer 编码器的堆叠层数。

3)文档级编码

使用 Transformer 编码器层对以标识符[CLS]为表征的句向量 h_i 进行文档级编码,记为 E_{Si}。添加位置嵌入 E'_i 以区分不同句子,并在嵌入序列前添加标识符 E_D 来表示整个文档。通过多层 Transformer 编码器的训练,得到文档 D 和每个句子 s_i 的文档级特征向量 $\boldsymbol{h}_D$ 和 $\boldsymbol{h}_{si}$。

4)冗余感知

使用文档级特征向量 $\boldsymbol{h}_D$ 和 $\boldsymbol{h}_{si}$ 之间的双线性匹配的评分函数得到句子的重要性得分。重要性得分 $F_{key}(s_i)$ 的计算方式如下:

$$F_{key}(s_i)=\frac{\exp h_D W_{ds} h_{si}}{\sum_{j=1}^{j=L}\exp h_D W_{ds} h_{sj}} \tag{1}$$

在每个第 l 步提取 n-$gram$ 匹配特征和语义匹配特征,得到候选句子的冗余特征。n-$gram$ 匹配特征 $f_{n\,gram}$ 的计算如下:

$$f_{n\text{-}gram}=\frac{|n\text{-}gram(\hat{S}_{l-1})\cap n\text{-}gram(s_i)|}{n\text{-}gram(s_i)} \tag{2}$$

其中,s_i 表示目前选定的句子 $\hat{S}_{l-1}$,n-$gram(x)$ 是连续 n 字的集合,$n=\{1,2,3\}$。

语义匹配特征 f_{sem} 的计算如下:

$$f_{sem}=\max_{\hat{s}_j\in\hat{S}_{l-1}}\cos(h_{si},h_{\hat{s}j}) \tag{3}$$

由于大多数余弦值都在接近 1 的小范围内,因此对 f_{sem} 进行放大得到 $\tilde{f}_{sem}$。

将 n-$gram$ 匹配特征和语义匹配特征连接得到整体冗余特征向量 $\boldsymbol{F}_{dun}(s_i)$:

$$\boldsymbol{F}_{dun}(s_i)=[f'_{1\text{-}gram};f'_{2\text{-}gram};f'_{3\text{-}gram};\tilde{f}'_{sem}] \tag{4}$$

其中,f' 是 f 经过二值化后的 one-hot 向量。

使用三维矩阵 W_F 在冗余特征 $\boldsymbol{F}_{dun}(s_i)$ 和重要性得分 $F_{key}(s_i)$ 之间进行双线性匹配,得到具有维数 d 的输出匹配向量并计算最终分数:

$$f(s_i)=W_f\tanh(F_{key}(s_i)\boldsymbol{W}_F\boldsymbol{F}_{dun}(s_i)) \tag{5}$$

1.2　摘要生成模型

选择双编码器-单解码器框架作为摘要生成模型的基本体系结构,同时将训练完毕的 BERT 编码器作为摘要生成模型的编码器。底层编码器间共享参数以减少内存和计算需求。

1)选择指导信号

使用 Oracle 贪婪算法[19]从原文中选择 N 个能最大化 ROUGE 得分的关键句作为指导信号,以充实指导信号包含的原文信息。

算法 1(Oracle 贪婪算法)

```
Input: A source document x, a reference summary y, a pre-defined integer N
Output: Oracle-selected important sentences r
    r={}
    for i=1,…,N do
      max_rouge=0
      for s in x/r do
        rouge_1, rouge_2=cal_rouge (r⋃s)
        cur_rouge=rouge_1+rouge_2
        if cur_rouge>max_rouge then
          max_rouge=cur_rouge
          sents_num=s
        end if
      end for
      r=r⋃{sents_num}
    end for
      return r
```

2)编码器

将由句子 $x_1,\cdots,x_{|X}$ 组成的原文 X 和由句子 $g_1,\cdots,g_{Y|}$ 组成的指导信号 G 分别进行输入表征,并送入两个编码器以获取相应的特征表示 x_i 和 g_i。编码器由 $n+1$ 个相同的编码器层堆叠而成。底层编码器结构如图 2 所示。

每个编码器层由一个多头自注意力模块 $SelfAttn$ 和一个前馈网络模块 FFN 组成:

$$\tilde{x}^l=LN(x^{l-1}+SelfAttn(x^{l-1})) \tag{6}$$

$$x^l=LN(\tilde{x}^{l-1}+FFN(\tilde{x}^{l-1})) \tag{7}$$

$$\tilde{g}^l=LN(g^{l-1}+SelfAttn(g^{l-1})) \tag{8}$$

$$g^l=LN(\tilde{g}^{l-1}+FFN(\tilde{g}^{l-1})) \tag{9}$$

其中,LN 表示层归一化,l 是 Transformer 编码器的堆叠层数。

通过将 Query 向量和 Key 向量相乘,自注意力模块表征输入部分字向量间的相似度,并对结果进行拼接来表示

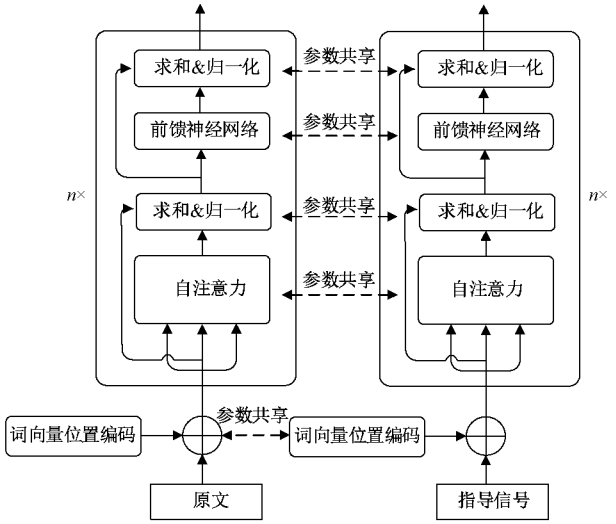

图 2　底层编码器结构

输入部分所有字向量间的权重和：

$$MH(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Concat(head_1,\cdots,head_h)\boldsymbol{W}^o \quad (10)$$

$$head_i = Attn(\boldsymbol{Q}\boldsymbol{W}_i^Q,\boldsymbol{K}\boldsymbol{W}_i^K,\boldsymbol{V}\boldsymbol{W}_i^V) \quad (11)$$

$$Attn(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = Softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}})\boldsymbol{V} \quad (12)$$

其中，$\boldsymbol{W}^o$ 为附加权重矩阵；$\boldsymbol{Q}$、$\boldsymbol{K}$ 和 $\boldsymbol{V}$ 分别为 Query 向量、Key 向量和 Value 向量；$Softmax$ 为归一化激活函数。

前馈网络模块包含两个线性变换和一个 ReLU 激活函数，能有效增加模型的非线性拟合能力：

$$FFN(\tilde{x}) = \max(0,\tilde{x}W_1 + b_1)W_2 + b_2 \quad (13)$$

其中，W_1 和 W_2 为线性变换，b_1 和 b_2 为偏置。

在编码过程中，原文与指导信号相互独立，两个编码器之间共享底层编码器和嵌入层的参数以减少内存需求，顶层编码器分别获取原文和指导信号的信息。

3)解码器

本文使用包含一个多头自注意力模块 $SelfAttn$、两个编码-解码注意力模块 $CrossAttn$ 和一个前馈网络模块 FFN 的 Transformer 解码器处理原文和指导信号的特征向量。解码器由 n 个相同的解码器层组成，结构如图 3 所示。

在进行解码操作时，使用编码-解码注意力模块处理指导信号特征向量 $\boldsymbol{g}$ 并生成相应的指导感知表示：

$$\tilde{y}^l = LN(y^{l-1} + SelfAttn(y^{l-1})) \quad (14)$$

$$\tilde{y}l = LN(\tilde{y}l-1 + CrossAttn(\tilde{y}l-1, g^{l-1})) \quad (15)$$

然后，解码器根据指导感知表示对原文特征向量 x 进行处理，并生成输出表示：

$$\tilde{y}l = LN(\tilde{y}l-1 + CrossAttn(\tilde{y}l-1, x^{l\ 1})) \quad (16)$$

最后，将输出表示送入前馈网络模块：

$$y^l = LN(\tilde{y}l-1 + FFN(\tilde{y}l-1)) \quad (17)$$

训练模型参数 θ 以最小化损失函数，使并行训练语料库 $\langle X,Y,G\rangle$ 输出的条件似然最大化：

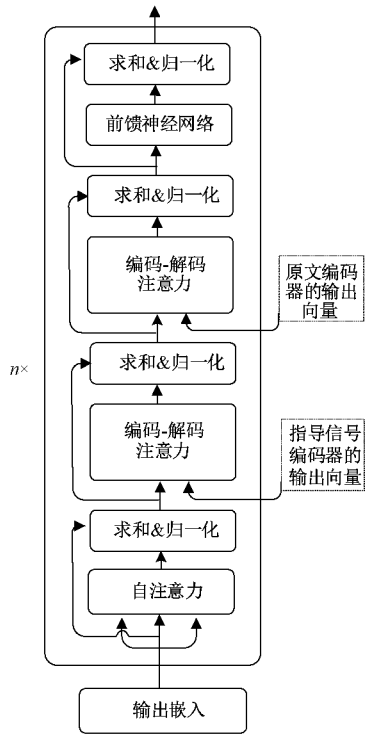

图 3　解码器结构

$$\hat{y} = \underset{\theta}{\operatorname{argmax}} \sum_{\langle x^i,y^i,g^i\rangle \in \langle X,Y,G\rangle} \log p(y^i \mid x^i, g^i;\theta) \quad (18)$$

4)二阶段微调

本文将训练完毕的 BERT 双编码器和随机初始化的 Transformer 解码器相结合构造生成模型，因此在生成阶段，编码器已经训练完毕，而解码器需要从随机初始化开始训练。训练的不平衡会导致编、解码器出现过拟合、欠拟合等问题。因此在生成阶段使用二阶段微调策略，将编码器和解码器的优化器分开，同时对编码器采用较小的学习率和平滑的衰减进行微调，使得模型在稳定训练解码器的同时使用更精确的梯度对编码器进行训练。

2　实验设置

2.1　实验数据集

本文的实验使用 Hu 等[21]提供的 LCSTS 中文短文本摘要评测数据集，其样本来自新浪微博的用户发表的微博内容。具体信息如表 1 所示。

表 1　LCSTS 数据集信息统计

	训练集	验证集	测试集
文本数量	2 400 591	10 666	1 106
平均文本字符	116	115	116
平均摘要字符	15	16	16

2.2　评价方法

为了获得客观、真实的评价结果，本文采用 ROUGE

评价方法与层次分析法互为补充。

1)ROUGE 评价方法

采用文本摘要领域的基准评价指标,即 Lin 等[22]提出的 ROUGE 自动摘要评价指标评价模型的性能。本文使用 ROUGE-N (N 为 1,2)和 ROUGE-L 进行评测。其中,ROUGE-N 计算模型生成摘要和参考摘要间重叠连续 N 字的词的召回率:

$$ROUGE\text{-}N=\frac{\sum_{S\in\{RS\}}\sum_{gram_n\in S}C_{match}(gram_n)}{\sum_{S\in\{RS\}}\sum_{gram_n\in S}C(gram_n)} \tag{19}$$

其中,$C(gram_n)$ 表示参考摘要中连续 N 字的词的数量;$C_{match}(gram_n)$ 表示参考摘要和模型生成摘要互相重叠的连续 N 字的词的个数。

2)层次分析法

使用 Delphi 专家调查法对生成摘要质量进行比较打分,采用层次分析法[23]对专家给出的定性结果进行定量分析。具体过程如下:

(1) 建立系统化的摘要质量评价分析的递阶层次结构模型。将可读性、精炼性、准确性和文学性作为模型生成摘要的质量评价因子。其中,可读性考察生成摘要是否流畅连贯;精炼性考察生成摘要是否包含重复文本;准确性考察生成摘要内容是否与原文事实一致;文学性考察生成摘要是否具有一定的文学美感。根据待评价模型和生成摘要的质量评价因子建立递阶层次结构模型,如图 4 所示。

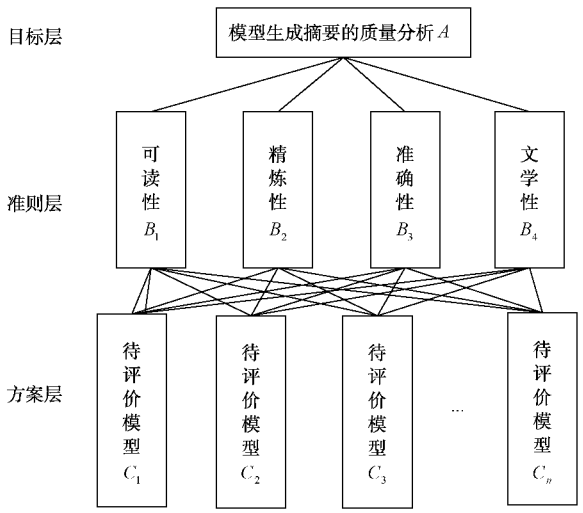

图 4　摘要质量评价分析的递阶层次结构模型

(2) 基于摘要质量评价体系,使用专家调查法构造判断矩阵。随机选择待评价模型生成的 100 条摘要建立测试集,采用专家调查问卷回收的方式,请 5 位中文系研究生依据表 2 判断矩阵标度及含义,对摘要质量进行评分。

(3) 计算各层次相对权重及一致性检验。使用几何平均法计算层次相对权重:

表 2　判断矩阵评价指标及含义

标度 a_{ij}	含义
1	i 和 j 的评价得分相同
3	i 比 j 的评价得分稍微高
5	i 比 j 的评价得分明显高
7	i 比 j 的评价得分非常高
9	i 比 j 的评价得分极端高
2,4,6,8	上述相邻判断的中间值

$$W_i=\frac{(\prod_{j-1}^{n}a_{ij})^{\frac{1}{n}}}{\sum_{i=1}^{n}(\prod_{j=1}^{n}a_{ij})^{\frac{1}{n}}},i=1,2,\cdots,n \tag{20}$$

然后计算一致性指标 CI:

$$CI=\frac{\lambda_{\max}-n}{n-1} \tag{21}$$

其中,$\lambda_{\max}$ 为所构建判断矩阵的谱半径。

最后,查平均随机一致性指标 RI 取值表,并计算判断矩阵的一致性比率:

$$CR=\frac{CI}{RI} \tag{22}$$

当 $CR<0.10$ 时,认为判断矩阵的设置较合理,不必对其做任何修改。

(4) 对各层次权重进行总排序操作。首先由高到低依次计算准则层和方案层的元素对目标层的合成权重:

$$CR=\frac{\sum_{i=1}^{n}b_iCI_i}{\sum_{i=1}^{n}b_iRI_i} \tag{23}$$

然后对各层次进行一致性检验,最后根据各分区的总权重值由大到小对待评价模型进行排序。

2.3　实验参数及设置

本文的 BERT 模型采用中文预训练模型 bert-base-chinese,设置最大序列长度为 512,解码时的 Beam_size 为 5。模型参数如表 3 所示。

表 3　BERT 预训练模型参数

参数	参数值
隐藏层数	12
注意力头数	12
嵌入层数	128
隐藏层单元数	768
词表大小	21 128

BERT 抽取式模型微调的训练步数 Training_steps 为 100 000,批处理大小 Batch_size 为 3 000,丢失率 Drop_out 为 0.1,学习率 Learning_rate 为 2×10^{-3},预热步骤参数 Warmup_steps 为 10 000。

BERT 生成式模型微调的训练步数 Training_steps 为 200 000，训练时 Batch_size 为 140。本文使用二阶段微调策略，表 4 为生成式摘要任务下编码器和解码器的参数对比。

表 4　编码器与解码器参数对比

参数值	编码器	解码器
Drop_out	0.2	0.2
Learning_rate	2×10^{-3}	0.1
Warm_steps	20 000	10 000

本文实验环境及配置如表 5 所示。

表 5　实验环境及配置

实验环境	实验配置
操作系统	Windows 10
编程语言	Python3.6
深度学习框架	Pytorch1.1.0
开发工具	PyCharm

3　实验结果与分析

3.1　基准实验对比分析

为验证本文模型处理文本摘要任务的优越性，选取 6 种基准模型在 LCSTS 数据集上进行对比实验。具体如下：

1) P-Gen[24]：该模型全称 Pointer Generator，使用指针网络结构结合 Coverage 机制减少了生成摘要中的重复词和未登录词，是生成式摘要的里程碑模型。

2) KBPM[25]：该模型使用 TextRank 从原文中抽取关键词，利用 BERT 预训练模型对其向量化并送入带有双重注意力机制的指针模型中得到最终的摘要。

3) GSum[20]：该模型为生成式模型，将原文和指导信号分别输入基于 Transformer 模型的双编码器-解码器结构，减少了生成摘要的事实性错误。

4) ARedSum[19]：该模型为抽取式模型，将 BERT 预训练模型和冗余感知方法相结合评估句子的重要性和冗余度，有助于减少摘要冗余。

5) PreSumm[18]：该模型将抽取式与生成式模型相融合，在传统端对端基础框架上，利用一个训练完整的抽取式模型优化生成式摘要效果。

6) TCAttn-GRU[26]：该模型采用基于卷积神经网络的双编码器对原文进行编码，将解码器同指针机制和集束搜索、相结合以提升摘要流畅度。

其中 * 表示论文中的数据。由表 6 可以看出：

1)本文模型与 P-Gen 模型相比，在 ROUGE 评价指标上取得较大提升，这是由于 P-Gen 模型不能充分聚焦原文关键信息，导致生成摘要存在事实错误，准确性不足。本文模型使用的双编码器-解码器结构能够全面地表征句子之间的关联，对于生成更贴近原文大意的摘要有着重要的指导作用。

表 6　基准模型 ROUGE 结果对比(%)

Model	ROUGE-1	ROUGE-2	ROUGE-L
P-Gen*	26.28	5.58	25.33
KBPM*	32.50	14.10	31.30
GSum	33.91	21.45	31.29
ARedSum	33.09	21.23	30.45
PreSumm	32.75	20.61	30.14
TCAttn-GRU*	33.20	24.10	31.50
HRAGS	34.28	22.16	31.44

2)本文模型比 KBPM 模型的 ROUGE 得分均有提升，说明相较于使用关键词辅助摘要生成的 KBPM 模型，本文使用 Oracle 贪婪算法生成的关键句包含更丰富的原文关键信息，且本文的双编码器-单解码器结构比 KBPM 模型的指针结构具备更强大的关键信息聚焦能力。

3)本文模型比 GSum 模型得分更高，说明本文模型在生成式模型 GSum 的基础上共享抽取式模型的参数更好地提升了生成摘要的效果，使用二阶段微调策略使得编、解码器稳定训练，抽取式模型编码的知识得到充分利用。

4)本文模型与 ARedSum 模型相比，ROUGE-2 和 ROUGE-L 分别提升了 0.93 和 0.99 个百分点，表明相比于单独的抽取式模型，本文使用混合式模型可以有效地结合抽取与生成方法的优点，生成可读性高且忠实于原文的摘要。

5)本文模型与混合式模型 PreSumm 相比，在 ROUGE-1、ROUGE-2 和 ROUGE-L 上分别提升 1.53、1.55 和 1.30 个百分点，说明本文在混合式模型基础上融入指导信号的方法是有效的，与 PreSumm 模型的单编码器相比，本文的双编码器能获取到更为全面的语义信息和更丰富的高层特征。

6)本文模型比 TCAttn-GRU 模型的 ROUGE-1 得分略有提升，因为 TCAttn-GRU 模型采用基于卷积神经网络双编码器对原文进行编码，本文模型使用具有更强大向量表征能力的 Transformer 模型作为双编码器的基础架构，得到更精确的词向量表示。

3.2　扩展实验对比分析

本文在测试阶段改变了 Transformer 编解码器层数、解码器的多头注意力数量和生成摘要的最短长度，分析其对模型效果的影响。

1)Transformer 层数对结果的影响

本文模型通过 Transformer 的加权平均来关注特定的重要信息片段，使得计算复杂度高并且耗时。逐层减少 Transformer 编解码器层数并衡量模型性能差异。实验结果表明，在测试期间仅改变编解码器层数不会对结果产生任何影响。

2)解码器的注意力数量对结果的影响

由于解码时的自回归特性，解码端的多头注意力耗时多于编码端，因此本文在保证注意力头数能被嵌入维度整除的基础上，逐个移除 Transformer 解码端的多头注意力，

进行测试并衡量模型性能差异。

表 7 结果表明,即使模型使用多个注意力头进行训练,在测试期间去除部分不会对性能产生负面影响,当进一步去除时,模型性能急剧下降。除了影响下游任务性能,去除多头注意力也能减少模型参数,提升测试速度。因此可在内存受限的平台部署经过合理剪枝的模型,在不影响模型效果的基础上减少训练时间。

表 7　不同多头注意力数量下的 ROUGE 结果对比

%

Num	ROUGE-1	ROUGE-2	ROUGE-L
1	19.42	8.32	17.97
2	23.63	11.87	21.46
4	26.77	14.13	24.15
6	32.25	19.86	29.28
8	33.87	21.78	30.69
12	33.89	20.83	30.37
16	33.21	20.89	30.44

3)生成摘要长度对结果的影响

设置合适的目标摘要长度是自动摘要中重要的问题之一。本文改变生成摘要的最短长度限制,分析模型生成摘要的长度对效果的影响。

表 8　不同摘要长度下的 ROUGE 结果对比　%

Min Length	ROUGE-1	ROUGE-2	ROUGE-L
10	33.28	21.39	30.42
11	33.57	21.60	30.68
12	33.87	21.78	30.69
13	34.28	22.16	31.44
14	34.25	21.94	30.71
15	33.63	21.36	30.37
16	33.47	21.20	30.16

表 8 结果表明,将生成摘要的最短长度分别设置为 13 和 14 个字符时,ROUGE 得分最高。在该条件下,生成摘要的平均长度更接近于参考摘要的平均长度,即 15～16 个字符。生成的摘要过短时无法包含原文主要概念,摘要过长则可能具有不忠实于原文的内容,都会导致摘要质量下降。

3.3　层次分析法对比分析

由图 5 可知,抽取式模型 ARedSum 的总权重值最低,说明抽取式摘要的可读性与精炼性存在不足。本文模型在 LCSTS 数据集上生成摘要的总权重值最高,说明本文混合式摘要模型的 BERT 双编码器对原文中的关键语义信息有了更全面且深层次的理解,具有双编码-解码注意力模块的 Transformer 解码器能有效融入指导信号的特征信息,生成内容丰富、描述准确、行文流畅的摘要。

3.4　摘要实例分析

由表 9 可知,本文模型生成了“嫌犯”、“逮捕”和“蓝翔挖掘机技术学校”等关键信息,可知本文模型充分发挥了二阶段微调和双编码-解码器的优势,抽取式模型进一步提升了生成式模型的性能,而不会产生累积误差,在指导信号的控制下,生成摘要内容紧扣文章中心且精简连贯。

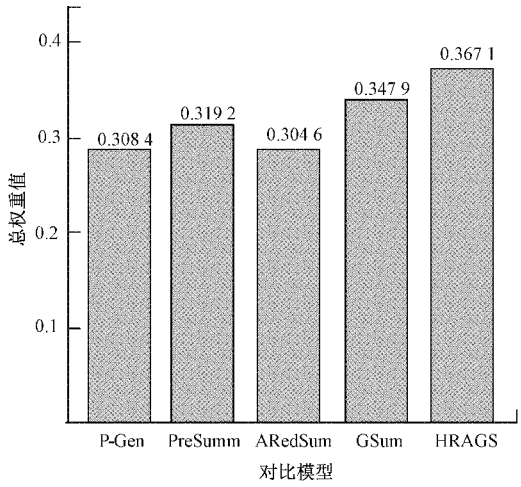

图 5　模型的总权重值对比

表 9　模型生成摘要示例

	类型	文本
例 1	原文	7 月 29 日,北京市大兴区摔死女童案犯罪嫌疑人韩某、李某分别因涉嫌故意杀人罪和窝藏罪,被依法批准逮捕。7 月 23 日,两名驾车男子因不满一名推着婴儿车的女士挡道与该女士发生争执,过程中一名男子将婴儿车内的女童摔在地上,导致女童死亡。
	参考摘要	北京摔死女童案疑犯被逮捕
	HRAGS	北京大兴摔死女童案嫌犯被逮捕
例 2	原文	中秋假期,刘先生到江西一风景区度假。当他们来到景区一书院办理入住登记时,工作人员让他们出示身份证与学历证书,并告知入住书院一定要是本科以上的学历。该景区表示书院是文化之地,为了给客人静养修学提供好的环境。出去谁带学历证明?
	参考摘要	游客没带本科文凭求宿遭拒
	HRAGS	江西一景区办理入住登记要求本科以上学历
例 3	原文	那么问题来了,挖掘机技术哪家强? 在蓝翔毕业生眼中,这家逆天的技术学校并没有那么强。来了工地才知道,其实这些实践操作还不如直接到工地拜师学,花费肯定比学校低! 蓝翔挖掘机专业毕业的学生表示。
	参考摘要	蓝翔技校毕业生:学挖掘机不如到工地拜师
	HRAGS	蓝翔挖掘机技术学校毕业生:不如拜师学

4　结　　论

本文提出了一种基于 HRAGS 模型的混合式摘要生成方法，使用双编码器-解码器结构和二阶段微调策略有效地指导模型生成更具可读性和准确性的摘要，减少了模型的累计误差。在 LCSTS 数据集上的实验结果和摘要实例都证明了本文模型的优势。但本文模型仍存在复杂度高、训练时间长等局限，在今后的研究中，我们将进一步对模型进行合理剪枝以减少模型的参数量和计算量。

参考文献

[1] JIA R, CAO Y, FANG F, et al. Deep differential amplifier for extractive summarization [C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021: 366-376.

[2] KWON J, KOBAYASHI N, KAMIGAITO H, et al. Considering nested tree structure in sentence extractive summarization with pre-trained transformer [C]. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Dominican Republic, 2021: 4039-4044.

[3] LIU Y, LIU P. SimCLS: A simple framework for contrastive learning of abstractive summarization[C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021:1065-1072.

[4] WU W, LI W, XIAO X, et al. BASS: Boosting abstractive summarization with unified semantic graph[C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021: 6052-6067.

[5] CHEN M, LI W, LIU J, et al. SgSum: Transforming multi-document summarization into sub-graph selection [C]. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Dominican Republic, 2021: 4063-4074.

[6] CAO S, WANG L. CLIFF: Contrastive learning for improving faithfulness and factuality in abstractive summarization [C]. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Dominican Republic, 2021: 6633-6649.

[7] MENDES A, NARAYAN S, MIRANDA S, et al. Jointly extracting and compressing documents with summary state representations[C]. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019: 3955-3966.

[8] BAE S, KIM T, KIM J, et al. Summary level training of sentence rewriting for abstractive summarization [C]. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019: 10-20.

[9] KRISHNA K, KHOSLA S, BIGHAM J, et al. Generating SOAP notes from doctor-patient conversations using modular summarization techniques [C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021: 4958-4972.

[10] SONG K, WANG B, FENG Z, et al. A new approach to overgenerating and scoring abstractive summaries [C]. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021: 1392-1404.

[11] MOROSHKO E, FEIGENBLAT G, ROITMAN H, et al. An editorial network for enhanced document summarization [C]. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019: 57-63.

[12] GAO S, CHEN X, LI P, et al. How to write summaries with patterns? Learning towards abstractive summarization through prototype editing[C]. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019: 3471-3751.

[13] SHAH D, YU L, LEI T, et al. Nutri-bullets hybrid: Consensual multi-document summarization [C]. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021: 5213-5222.

[14] ZHU F, TU S, SHI J, et al. TWAG: A topic-guided wikipedia abstract generator[C]. Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, 2021: 4623-4635.

[15] SAJAD S, NAZLI J, ROSS W. Attend to medical ontologies: Content selection for clinical abstractive

summarization[C]. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020: 1899-1905.

[16] 王义真,欧石燕,陈金菊. 民事裁判文书两阶段式自动摘要研究[J]. 数据分析与知识发现,2021,5(5):104-114.

[17] EGONMWAN E, CHALI Y. Transformer-based model for single documents neural summarization[C]. Proceedings of the 3rd Workshop on Neural Generation and Translation, 2019: 70-90.

[18] LIU Y, LAPATA M. Text summarization with pretrained encoders [C]. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing,2019: 3730-3740.

[19] BI K, JHA R, CROFT B, et al. AREDSUM: Adaptive redundancy-aware iterative sentence ranking for extractive document summarization [C]. Association for Computational Linguistics, 2021:281-291.

[20] DOU Z Y, LIU P, HAYASHI H, et al. GSum: A general framework for guided neural abstractive summarization [C]. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021: 4830-4842.

[21] HU B, CHEN Q, ZHU F. LCSTS: A large scale Chinese short text summarization dataset[C]. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015: 1967-1972.

[22] LIN C Y. ROUGE: A package for automatic evaluation of summaries[C]. In Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004),2004.

[23] MOHER D, COOK D J, EASTWOOD S, et al. Improving the quality of reports of meta-analyses of randomised controlled trials: The QUOROM statement. quality of reporting of meta-analyses[J]. Br J Surg, 2000, 23(6): 1448-1454.

[24] SEE A, LIU P J, MANNING C D. Get to the point: Summarization with pointer-generator networks[C]. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, 2017: 1073-1083.

[25] 李伯涵,李红莲. 一种融合关键词的生成式摘要方法[J/OL]. 计算机应用研究: 1-5[2021-11-17]. https:. doi. org/10. 19734/j. issn. 1001-3695. 2021. 04. 0111.

[26] 高巍,马辉,李大舟,等. 基于双编码器的中文文本摘要技术的研究与实现[J]. 计算机工程与设计,2021,42(9): 2687-2695.

作者简介

岳琳,硕士研究生,主要研究方向为自然语言处理。
E-mail: ylajhb@163. com

杨风暴(通信作者),博士,教授,主要研究方向为自然语言处理、数据融合等。

王肖霞,博士,副教授,主要研究方向为自然语言处理、关联成像等。