计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2024,33(5):210–217 [doi: 10.15888/j.cnki.csa.009508]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于实体复制和双粒度指导的抽象摘要①

周子力, 高士亮, 安润鲁, 包新月

(曲阜师范大学 网络空间安全学院, 曲阜 273165)
通信作者: 周子力, E-mail: zzl@qfnu.edu.cn

摘　要: 抽象神经网络在文本摘要领域取得了长足进步, 展示了令人瞩目的成就. 然而, 由于抽象摘要的灵活性, 它很容易造成生成的摘要忠实性差的问题, 甚至偏离源文档的语义主旨. 针对这一问题, 本文提出了两种方法来提高摘要的保真度. (1) 由于实体在摘要中起着重要作用, 而且通常来自于原始文档, 因此本文提出允许模型从源文档中复制实体, 确保生成的实体与源文档中的实体相匹配, 这有助于防止生成不一致的实体. (2) 为了更好地防止生成的摘要与原文产生语义偏离, 本文在摘要生成过程中使用关键实体和关键 token 作为两种不同粒度的指导信息以指导摘要的生成. 本文使用 *ROUGE* 指标在两个广泛使用的文本摘要数据集 CNNDM 和 XSum 上评估了本文方法的性能, 实验结果表明, 这两种方法在提高模型性能方面都取得了显著的效果. 此外, 实验还证明了实体复制机制可以在一定程度上借助指导信息以纠正引入的语义噪声.

关键词: 抽象摘要; 实体复制; 双粒度指导; 深度学习; 预训练模型

引用格式: 周子力,高士亮,安润鲁,包新月.基于实体复制和双粒度指导的抽象摘要.计算机系统应用,2024,33(5):210–217. http://www.c-s-a.org.cn/1003-3254/9508.html

Abstractive Summarization Based on Entity Copy and Dual Granularity Guidance

ZHOU Zi-Li, GAO Shi-Liang, AN Run-Lu, BAO Xin-Yue

(School of Cyber Science and Engineering, Qufu Normal University, Qufu 273165, China)

Abstract: Abstract neural networks have made significant progress and demonstrated remarkable achievements in the field of text summarization. However, abstract summarization is highly likely to generate summaries of poor fidelity and even deviate from the semantic essence of the source documents due to its flexibility. To address this issue, this study proposes two methods to improve the fidelity of summaries. For Method 1, since entities play an important role in summaries and are usually derived from the original documents, the paper suggests allowing the model to copy entities from the source document to ensure that the generated entities match those in the source document and thereby prevent the generation of inconsistent entities. For Method 2, to better prevent the generated summary from deviating from the original text semantically, the study uses key entities and key tokens as two types of guiding information at different levels of granularity in the summary generation process. The performance of the proposed methods is evaluated using the *ROUGE* metric on two widely used text summarization datasets, namely, CNNDM and XSum. The experimental results demonstrate that both methods have significantly improved the performance of the model. Furthermore, the experiments also prove that the entity copy mechanism can, to some extent, use guiding information to correct introduced semantic noise.

Key words: abstract summarization; entity copy; dual granularity guidance; deep learning; pre-train model

① 基金项目: 山东省自然科学基金 (ZR2021MD115); 上海市科委项目 (21511100302)
收稿时间: 2023-12-12; 修改时间: 2024-01-10; 采用时间: 2024-01-17; csa 在线出版时间: 2024-04-01
CNKI 网络首发时间: 2024-04-03

文本摘要任务旨在保留原始源文本语义的前提下, 将文档或文档集浓缩成简明的摘要[1]. 文本摘要技术通常分为抽取式和抽象式, 抽取式是从输入文本中选取重要的词或者句子, 然后将其合并成摘要[2–4]; 抽象式则是通过模型理解原文语义而后重新组织自主生成摘要的方法[5,6]. 与抽取式相比, 抽象式具有更大的灵活性, 更有可能生成准确度高的摘要.

典型的抽象摘要的算法使用带有注意力机制[7]、复制机制[8,9]、信息指导[10,11]或者知识增强的序列到序列的模型. 然而, 抽象摘要的灵活性也使其存在更多问题. 一是产生忠实性差的摘要, 包括关键内容不一致和关键语义与主旨偏离的问题. 二是由于在语义收集过程中无法准确地学习源文档的语义信息, 导致语义学习不足或者语义偏差, 进而无法生成更好的摘要. 针对第 1 个问题, 本文提出允许模型复制源文件中实体的策略, 这样可以有效防止关键信息出错, 确保摘要的质量. 针对第 2 个问题, 本文提出使用不同粒度的指导信息指导摘要的生成, 这样可以有效控制学习到的语义特征, 保证摘要的忠实性.

在本文中, 提出了 CE2GSum 模型, 这是一个利用实体复制和双粒度引导进行联合训练的框架. 具体而言, 本文首先从源文档中抽取实体 (实体短语), 这使得模型在解码阶段能够复制实体, 可以针对重要的实体直接复制而不是通过一个个的 token 生成, 这样可以有效保证关键信息不会出错. 此外, 还将提取的实体视为候选实体, 并计算它们在源文档中的相关性, 以获得关键实体, 这些关键实体将用作多粒度级的指导信息. 同时, 还确定源文档中的关键 token, 这些 token 将用作单粒度级别的指导信息. 通过在生成过程中利用两个不同粒度级别的指导信息, 结合实体复制, 进而可以生成可靠的摘要.

本文对两个基线文本摘要数据集 (XSum[12]和 CNNDM[13]) 进行的评估实验, 实验表明, 本文的 CE2GSum 模型在文本摘要指标上都获得了提升, 说明模型在保真度上表现优秀. 结果明确展示了我们的模型在生成准确且主旨一致于源文本的摘要方面取得的显著效果.

1 相关工作

1.1 抽象摘要

Rush 等人在 2005 年首次基于注意力机制的序列到序列的模型用于生成摘要[14], 自此以后编码器-解码器架构在文本摘要领域被广泛地应用. 2017 年, Vaswani 等人提出了划时代模型 Transformer[15], 以此取代传统的循环神经网络, 解决了长序列输入的问题. 随后, 预训练模型被引入抽象摘要任务中, 2019 年, Dong 等人首次利用 BERT 作为编码器[3], 以大型预训练捕捉源文档的丰富语义信息, 而解码器则依然使用 Transformer 进行解码, 在抽象摘要领域也取得了很好的性能; 2019 年, Lewis 等人提出了另一种预训练模型 BART[16], 随后 Raffel 在 2020 年提出了 T5[17], 这两种预训练方式不同于 BART, 使用 Transformer 的编码器和解码器同时训练得到的. 2020 年, Zhang 等人提出了 PEGASUS[18], 专门用于生成摘要的领域预训练模型, 自此研究者不断地在这些预训练模型上进行微调和改进, 以此获得适应于不同场景摘要任务的模型.

1.2 信息指导

由于基于预训练模型的摘要模型本身的复杂度, 很容易出现错误, 导致摘要不准确. 具体而言, 生成的摘要与源文档主旨不一致, 甚至有时与源文档语义矛盾[19,20]. 信息指导机制为解决这个问题也被提出并广泛使用, 2020 年, Dou 等人提出了使用关键词和关键句子作为指导信息的 GSum 模型, 并获得了不错的效果[21]. 2022 年, Xu 等人提出了使用关键短语作为指导信息的 GISG 模型[11]; 2021 年, Ma 等人提出了使用源文档主题信息作为摘要生成的指导信息的 T-BERTSum 模型[22]. 随后, 研究人员不断将各种类型的信息以不同的方式作为指导信息, 并取得了很大的提高. 基于以上, 本文提出了使用关键实体和关键 token 作为两种不同粒度的指导信息, 并且以不同于以上几种模型的方式在模型上融入信息指导机制.

1.3 实体复制

为了解决 OOV 问题, 即词汇表不足的问题, 2017 年, See 等人首次提出了指针生成网络 (pointer-generator network), 即允许模型从源文档中复制某些摘要需要的而词汇表中不存在的 token[23]. 2021 年, Liu 等人提出了 BioCopy 模型[24], 这是一种即插即用的架构, 它使用的是 BIO 和 token 的联合训练. 2021 年, Li 等人提出了利用历史分布预测当前分布的模型[25]. 上述大部分模型都是基于 token 级复制的, 而 2022 年提出了 SpanCopy 模型[26], 其允许模型复制实体以实现更好性能的方法.

2 CE2GSum 模型

正如图 1 所示 (左边部分为 token 生成的过程, 右

边部分为实体复制的过程.), 本文的模型也是基于序列到序列的模型的. 其中, 使用 PEGASUS 作为基础模型, 然后融合了实体复制模块和双粒度指导模块. 在第 2.1 节, 第 2.2 节, 第 2.3 节将分别介绍编码器和解码器、双粒度指导、实体解码器.

2.1 编码器解码器

本文使用 PEGASUS 预训练模型作为编码器和解码器, 并作为本文模型的基础架构. 抽象文本摘要可以描述为输入源文档 $X=\{X_1,X_2,\cdots,X_n\}$, 然后压缩为简洁的一段文本 $Y=\{Y_1,Y_2,\cdots,Y_m\}$.

$$H^e = Encoder(X) \tag{1}$$

令 $H^e=\{h_1^e,h_2^e,\cdots,h_n^e\}$, 其中, $h_i^e\in\mathbb{R}^h$, 表示第 i 个 token 的隐藏表示, h表示隐藏状态的维度大小. 在解码的第 t 步, 解码器通过利用编码器的上下文表示 H^e 和前缀词 $\{Y_1,Y_2,\cdots,Y_{t-1}\}$ 的编码器-解码器注意力生成第 t 个的隐藏表示 h_t^d. 最后确定从词汇表 V 中预测的 Y_t 的概率:

$$P(Y_t|Y_{<t},X) = Softmax(Eh_t^d) \tag{2}$$

其中, $E\in\mathbb{R}^{|V|\times h}$.

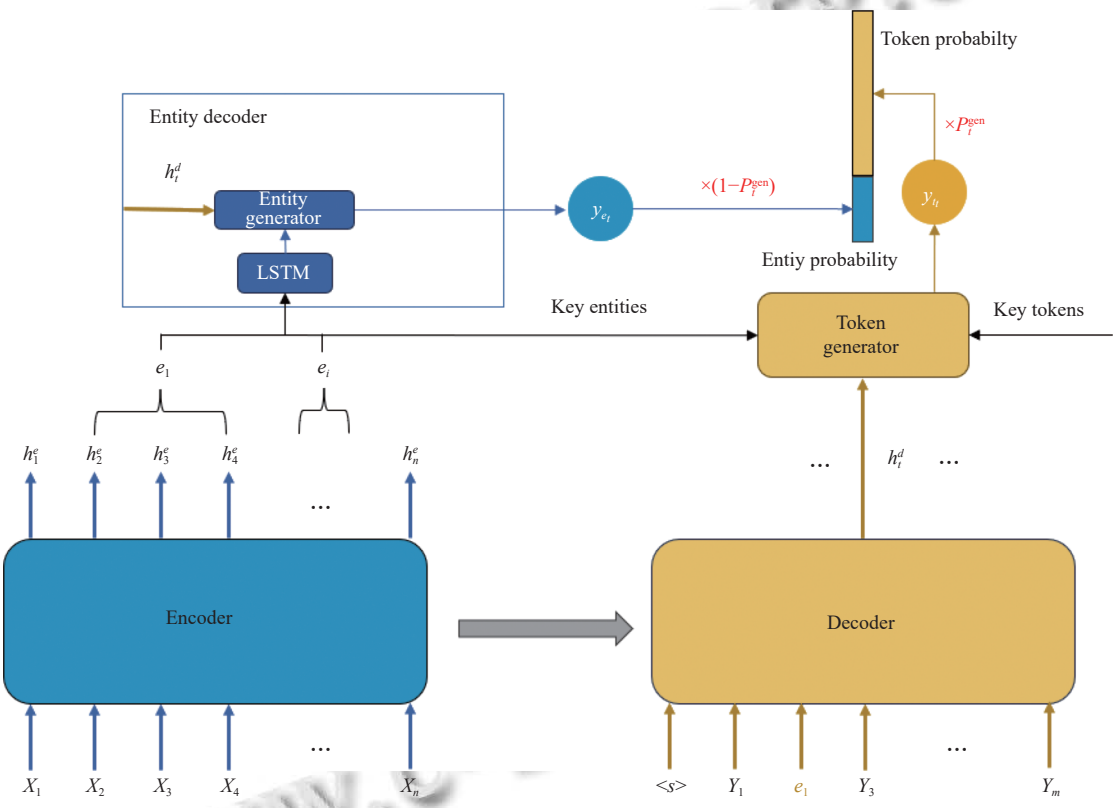

图 1 CE2GSum 模型

2.2 双粒度指导

计算关键 token 和关键实体信息的方法不同于直接从源文档中通过某些算法选取其中重要的关键 token 和关键实体. 相反, 本文则是根据每个 token 或者实体信息在词汇表上分配相关性概率分布.

2.2.1 关键 tokens

如图 2 所示, 使用上下文表示、嵌入矩阵、以及编码器和解码器的交叉注意力共同生成关键 token 指导信息. 其中, crossAttn表示编码器和解码器的交叉注意力, Embedding matrix表示嵌入矩阵.

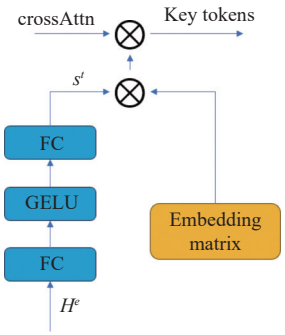

图 2 关键 token 的计算示意图

第 1 步: 首先使用编码器的隐藏表示来学习 token-vocabulary 的分布.

$$S^t = GELU(H^e W_1)W_2 E \tag{3}$$

其中, W_1和W_2是学习权重. H^e表示为编码器的隐藏上下文表示, E为嵌入矩阵. $S^t \in \mathbb{R}^{n\times|V|}$, 其中$\{s_1^t, s_2^t, \cdots, s_n^t\}$, $s_i^t \in \mathbb{R}^{|V|}$, 表示 token X_i和词汇表的 token 的上下文相似度.

第 2 步: 然后, 我们将 token-vocabulary 分布和解码器注意力结合, 最终生成指导信息. 使用注意力权重$a_t^{d,L}$对 token-vocabulary 相似度进行加权求和, 从而得到 token 级指导信息.

$$f_t^{\text{token}} = \sum_{i=1}^{n} a_{t,i}^{d,L} \cdot s_i^t \tag{4}$$

其中, $a_t^{d,L} \in \mathbb{R}^n$, 表示输出 token Y_t在最后一层解码器层L中与上下文表示之间的交叉注意力分布.

最后, 修改预测概率为:

$$p(Y_t|Y_{<t}, X) = Softmax(Eh_t^d + f_t^{\text{token}}) \tag{5}$$

2.2.2 关键实体

首先为了获取实体, 我们使用现有的命名实体识别工具 Spacy 提取实体. 每一个实体由$\{X_i, \cdots, X_j\}$组成, 则每个实体的表示为$e_i = avg(\{X_i, \cdots, X_j\})$.

相似地, 计算关键实体信息使用类似于关键 token 的计算方法, 计算过程如图 3 所示. crossAttn表示实体上下文和解码器的交叉注意力, Embedding matrix表示嵌入矩阵. 首先, 我们通过实体隐藏上下文获得 Entity-EntitiesSet 分布, 计算如式 (6), 式 (7) 所示:

$$H^{e\_\text{entity}} = LSTM(R) \tag{6}$$

其中, $R = \{e_1, e_2, \cdots, e_k\}$, 其中$k$表示实体的数量, $H^{e\_\text{entity}} = \{h_1^{e\_\text{entity}}, h_2^{e\_\text{entity}}, \cdots, h_k^{e\_\text{entity}}\}$.

$$S^e = (GELU(H^{e\_\text{entity}} W_3)W_4 E) \tag{7}$$

其中, $S^e = \{s_1^e, s_2^e, \cdots, s_k^e\}$, 其中$s_i^e \in \mathbb{R}^{|V|}$.

令$a_t^E \in \mathbb{R}^k$表示解码器在解码的第t步中实体上下文与解码器之间的交叉注意力分布. 然后使用a_t^E对 Entity-EntitiesSet 分布就行加权求和, 得到第t步解码的实体级指导信息$f_n^{\text{entity}} \in \mathbb{R}^{|V|}$.

$$f_t^{\text{entity}} = \sum_i a_{t,i}^E \cdot s_i^e \tag{8}$$

最终可以得到融合了 token 级指导信息和实体级指导信息的概率分布为:

$$p(Y_t|Y_{<t}, X) = Softmax((1-\lambda)h_t^d E + \lambda(f_t^{\text{token}} + f_t^{\text{entity}})) \tag{9}$$

其中, λ为超参数.

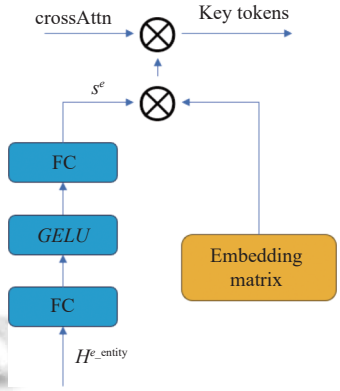

图 3 关键实体计算示意图

2.3 实体解码器

在每个解码步骤t中 (在图 1 左上部分), 作为实体解码器, 用于生成实体的逻辑向量, 同时, 还需要计算当前步骤的实体复制和 token 生成的概率.

2.3.1 实体生成

首先, 使用 LSTM 模型获取实体的上下文语义.

$$H^{e\_\text{entity}} = LSTM(R) \tag{10}$$

然后我们计算当前步的实体逻辑向量:

$$y_{e_t} = Q(h_t^d)K(h_j^{e\_\text{entity}}),\ y_{e_t} \in \mathbb{R}^k \tag{11}$$

其中, k表示实体的数量.

2.3.2 复制概率

将一个映射函数应用于隐藏状态, 将其映射为一个单一的表示, 即可作为复制实体的概率, 即:

$$p_t^{\text{copy}} = Sigmoid(FFN(h_t^d)),\ p_t^{\text{copy}} \in [0,1] \tag{12}$$

其中, p_t^{copy}就是代表第t步中选择复制实体的概率, 那么$p_t^{\text{gen}} = 1 - p_t^{\text{copy}}$是在第$t$步生成一个来自词汇表的 token 的概率. 最终的概率表示为:

$$p_t^{\text{final}} = [(1-p_n^{\text{gen}}) \times y_{e_t}, p_t^{\text{gen}} \times y_{t_t}] \tag{13}$$

其中, $p_t^{\text{final}} \in \mathbb{R}^{|V|+E}$, $|E|$是实体的数量. 并且$y_{t_t} = Softmax((1-\lambda)h_t^d E + \lambda(f_t^{\text{token}} + f_t^{\text{entity}}))$, f_t^{token}表示的是当前步生成 token 的逻辑向量.

2.4 损失函数

本文使用交叉熵损失函数作为模型最后的损失函数, 即:

$$Loss=\sum_{t}^{n}CrossEntropyLoss(p_t^{\text{final}},t_t) \tag{14}$$

因此, 真实标签可以是从词汇表中生成单词的索引, 也可以是从源文档复制的实体的索引, 即, $t_t\in[0,|V|+|E|]$, 其中, $|V|$表示词汇表的大小, $|E|$表示实体的数量.

3 实验分析

3.1 数据集

本文选取两个文本摘要通用数据集 XSum 和 CNNDM 上进行实验, 其中 CNNDM 来源于新闻媒体, 包含多个参考摘要, 而 XSum 则来自于新闻网站, 只有一个摘要. 为了获得过滤数据集, 使用工具 Spacy 抽取文章的实体, 随后, 对数据集进行过滤, 保留只包含那些文章摘要和源文档都存在相同实体的样本. 过滤后的数据集的详细统计如表 1 所示.

表 1 过滤和未过滤数据集的统计数据 (训练集/验证集/测试集)

数据集	未过滤	已过滤
CNNDM	287 111/13 368/13 365	105 847/4 490/3 903
XSum	204 017/11 327/11 333	4 2481/2 349/2 412

3.2 评估指标

ROUGE-3 被广泛用于自动摘要生成任务的评估[27], 因此在实验中使用 *ROUGE*-1 (*R*-1), *ROUGE*-2 (*R*-2), *ROUGE*-*L* (*R*-*L*) 作为评估指标. *R*-1 指的是生成的摘要与参考摘要之间重叠词的比例. *R*-2 衡量长度为 2 的连续词序列的重叠度. *R*-*L* 考虑生成摘要和参考摘要之间最长的共同子序列.

R-*N* 计算为:

$$R\text{-}N=\frac{\sum_{s\in\{ReferenceSummaries\}}\sum_{gram_n\in S}Count_{match}(gram_n)}{\sum_{s\in\{ReferenceSummaries\}}\sum_{gram_n\in S}Count(gram_n)} \tag{15}$$

其中, N的值分别为 1 和 2, 主要为了统计在 1-gram 和 2-gram 的召回率, 反映的是生成摘要 1 个粒度和 2 个粒度的重叠率, 进而表示摘要的准确程度. 对于 *R*-*L* 计算如下:

$$R_{LCS}=\frac{LCS(C,S)}{len(C)} \tag{16}$$

$$P_{LCS}=\frac{LCS(C,S)}{len(S)} \tag{17}$$

$$R\text{-}L=F_{LCS}=\frac{(1+\beta^2R_{LCS}P_{LCS})}{R_{LCS}+\beta^2P_{LCS}} \tag{18}$$

其中, S 表示人工摘要, 即真实摘要; C表示机器摘要, R_{LCS} 表示召回率, P_{LCS} 表示精确率, F_{LCS} 就是 *R*-*L* 的得分, 其主要在一定程度上可以体现生成的摘要的可读性.

3.3 实验细节

本文的实验在过滤数据集和原始数据上进行的, 使用单张 V100 GPU 作为实验设备, 使用 PEGASUS 作为模型的编码器和解码器, batch_size 设置为 6, 同时经过多次实验, 确定了超参数λ的取值为 0.2 时效果明显. 过滤 CNNDM 数据集的训练时间为 36 h, XSum 数据集的训练时间为 24 h. 在未过滤的 CNNDM 数据集, 训练时间为 60 h, XSum 数据集训练时间为 48 h.

3.4 结果分析

(1) 首先对过滤后的数据集进行了对比实验, 以评估模型的性能. 具体而言, 比较了基线模型 PEGASUS 和 SpanCopy 的性能, 比较的结果如表 2 所示. 在过滤数据集上进行评估时, 本文的模型在两个数据集的大多数指标上都比其他两个模型表现更优. 这表明本文的模型在摘要生成由于其他两个模型, 尤其是在 XSum 数据集上, 它在所有 3 个指标上都超过了其他两个模型, 其中, 本文模型的 *R*-2 和 *R*-*L* 指标都比其他两个模型高出约 0.9–1. 同时, 也证明我们引入的实体复制机制、双粒度信息指导机制可以提高模型性能.

表 2 过滤数据集上模型性能比较 (%)

模型	CNNDM			XSum		
	R-1	*R*-2	*R*-3	*R*-1	*R*-2	*R*-3
PEGASUS	44.70	22.23	32.52	43.01	19.00	34.01
SpanCopy	45.46	23.12	33.48	44.23	19.90	35.50
CE2GSum (our)	45.72	23.22	33.47	44.66	20.91	36.41

(2) 同样地, 也在未经过滤的数据集进行了实验, 其结果如表 3 所示. 可以得出, 本文的模型在 CNNDM 数据集上的表现并不令人满意, 分析主要是由于实体复制机制会引入的大量不必选择的实体噪声, 影响了模型性能. 然而, 在 XSum 数据集上却表现良好, 这归因于该数据集相对于 CNNDM 更具有抽象性的特性.

通过以上对过滤数据集和未过滤数据集的实验, 可以观察到本文的模型在未过滤数据集上表现差, 然而在过滤的数据集上却表现良好. 这种性能差异原因是未过滤数据集摘要中存在的实体并非直接来源于源

文件, 这就导致这些实体在模型生成过程中引入误导性信息, 致使性能指标下降. 然而, 本文的模型在抽象性强的 XSum 数据集上表现较好的结果, 可以看出, 尽管通过实体复制引入了噪声的挑战, 模型仍然在 XSum 性能良好, 这表明模型中的信息引导机制在一定程度上可以弥补实体复制机制引入的错误.

表 3 未过滤数据集上模型性能比较 (%)

模型/数据集	CNNDM			XSum		
	R-1	*R*-2	*R*-3	*R*-1	*R*-2	*R*-3
Pointer+Coverage +EntailmentGen+ QuestionGen	39.81	17.64	36.54	—	—	—
PEGASUS	44.62	20.82	31.05	46.65	23.47	38.67
BART	44.16	21.82	40.90	45.14	22.27	37.25
GSum	45.94	22.32	42.48	45.40	21.89	36.67
SpanCopy	44.16	20.61	30.97	46.23	22.76	37.96
CE2GSum	44.25	20.79	31.07	46.23	22.28	38.05

3.5 消融实验

在消融实验中, 主要关注点是评估使用过滤数据集的不同模块对模型性能的影响, 因此, 使用过滤后的 CNNDM 数据集进行消融实验分析, 以展示引入的实体复制机制和双粒度指导信息的有效性.

如表 4 所示, 本文的模型进行了 3 个消融实验, 这些实验移除不同的模块进行实验: 删除双粒度指导信息模块 (CE2GSum–(KT and KE))、删除关键 token (CE2GSum–(KT)) 和删除关键实体模块 (CE2GSum–(KE)), 以此分析不同模块对模型性能的影响.

表 4 在过滤 CNNDM 数据集上消融实验结果 (%)

数据集	*R*-1	*R*-2	*R*-*L*
PEGASUS	44.70	22.23	32.52
CE2GSum–(KT and KE)	45.44	22.99	33.15
CE2GSum–(KT)	45.48	22.89	33.17
CE2GSum–(KE)	45.53	23.24	33.51
CE2GSum	45.72	23.22	33.47

从表 4 中结果可以得出, 实体复制机制和双粒度引导机制均可以在不同程度上提高模型的表现性能. 其中, 实体复制机制略比双粒度引导机制更有效, 实体复制机制使指标提升了约 0.6–0.7, 而双粒度引导机制只提高了约 0.2–0.3. 此外, 同时包含关键实体和关键 token 指导信息也可以在不同程度上增强模型的性能.

4 结论与展望

4.1 结论

本文通过实验证明了 CE2GSum 模型的有效性. 通过对不同数据集进行实验以及在 CNNDM 数据集上的消融实验, 可以得出以下结论.

(1) CE2GSum 模型更适合过滤数据集的一个主要原因是, 在给定的未过滤数据集中存在大量摘要中提到的实体并非来自源文件的样本. 这会导致引入大量的噪声, 从而对模型的性能产生负面影响. 通过使用过滤数据集, 可以确保摘要中提到的实体来自源文件, 从而提高性能和结果的准确性.

(2) 实体复制机制和双粒度的指导信息都可以在不同程度上提高模型的性能, 实体复制对模型性能的提高略大于指导信息.

(3) 双粒度指导信息在一定程度上可以弥补实体复制机制带来的语义偏差, 并有效控制生成的摘要的语义.

4.2 展望

本文提出了一种新颖的抽象摘要生成模型, 具体来说, 在序列到序列模型架构的基础上, 首先引入了实体复制机制, 使得模型在生成 token 的基础上也可以复制源档中的实体, 通过实验, 验证了该机制的有效性; 此外, 在摘要生成过程中结合关键实体级 (多粒度) 和关键 token 级 (单粒度) 信息作为指导信息, 同样可以提高模型摘要生成的性能. 最后将两者公共引入模型, 并取得了显著的效果. 在未来的工作中, 一方面目标是提取更多的短语信息, 以此扩展模型在生成摘要时短语的可选项, 这样就可以将使模型在摘要生成过程中既能生成单个 token, 也能生成短语, 从而使模型更贴近人类实践. 另一方面, 为了生成摘要更加简洁, 贴合主旨, 在后续的工作中, 将使模型生成多个摘要, 然后通过另外一个小模型进行二次训练, 进而获得更加符合主旨的摘要.

参考文献

1 Chowdhury T, Kumar S, Chakraborty T. Neural abstractive summarization with structural attention. Proceedings of the 29th International Joint Conference on Artificial Intelligence. Yokohama: IJCAI.org, 2021. 514.

2 Zheng CJ, Zhang KP, Wang HJ, *et al*. Topic-aware abstractive text summarization. arXiv:2010.10323, 2020.

3 Dong L, Yang N, Wang WH, *et al*. Unified language model pre-training for natural language understanding and generation. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Vancouver:

Curran Associates Inc., 2019. 1170.
4 Jia RP, Cao YA, Tang HZ, *et al*. Neural extractive summarization with hierarchical attentive heterogeneous graph network. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2020. 3622–3631.
5 Zou YY, Zhang X, Lu W, *et al*. Pre-training for abstractive document summarization by reinstating source text. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2020. 3646–3660.
6 Li PJ, Lam W, Bing LD, *et al*. Deep recurrent generative decoder for abstractive text summarization. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen: Association for Computational Linguistics, 2017. 2091–2100.
7 Chowdhury T, Kumar S, Chakraborty T. Neural abstractive summarization with structural attention. Proceedings of the 29th International Joint Conference on Artificial Intelligence and the 17th Pacific Rim International Conference on Artificial Intelligence. IJCAI.org, 2020. [doi: 10.24963/ijcai.2020/510]
8 Xu S, Li HR, Yuan P, *et al*. Self-attention guided copy mechanism for abstractive summarization. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 1355–1362.
9 Liu Y, Lapata M. Text summarization with pretrained encoders. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong: Association for Computational Linguistics, 2019. 3730–3740.
10 Genest PE, Lapalme G. Fully abstractive approach to guided summarization. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers. Jeju Island: ACM, 2012. 354–358.
11 Xu W, Zhao TJ. Jointly learning guidance induction and faithful summary generation via conditional variational autoencoders. Proceedings of the 2022 Findings of the Association for Computational Linguistics. Seattle: Association for Computational Linguistics, 2022. 2340–2350.
12 Narayan S, Cohen SB, Lapata M. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels: Association for Computational Linguistics, 2018. 1797–1807. [doi: 10.18653/v1/D18-1206]
13 Riloff E, Chiang D, Hockenmaier J, *et al*. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels: Association for Computational Linguistics, 2018.
14 Rush AM, Chopra S, Weston J. A neural attention model for abstractive sentence summarization. Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon: Association for Computational Linguistics, 2015. 379–389.
15 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000–6010.
16 Lewis M, Liu YH, Goyal N, *et al*. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2020. 7871–7880.
17 Raffel C, Shazeer N, Roberts A, *et al*. Exploring the limits of transfer learning with a unified text-to-text Transformer. The Journal of Machine Learning Research, 2020, 21(1): 140.
18 Zhang JQ, Zhao Y, Saleh M, *et al*. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. Proceedings of the 37th International Conference on Machine Learning. JMLR.org, 2020. 1051.
19 Zhou CT, Neubig G, Gu JT, *et al*. Detecting hallucinated content in conditional neural sequence generation. Proceedings of the 2021 Findings of the Association for Computational Linguistics. Association for Computational Linguistics, 2021. 1393–1404.
20 Gabriel S, Celikyilmaz A, Jha R, *et al*. GO FIGURE: A meta evaluation of factuality in summarization. Proceedings of the 2021 Findings of Association for Computational Linguistics. Association for Computational Linguistics, 2021. 478–487.
21 Dou ZY, Liu PF, Hayashi H, *et al*. GSum: A general framework for guided neural abstractive summarization. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2021. 4830–4842.
22 Ma TH, Pan Q, Rong H, *et al*. T-BERTSum: Topic-aware

text summarization based on BERT. IEEE Transactions on Computational Social Systems, 2022, 9(3): 879–890. [doi: 10.1109/TCSS.2021.3088506]

23 See A, Liu PJ, Manning CD. Get to the point: Summarization with pointer-generator networks. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: Association for Computational Linguistics, 2017. 1073–1083.

24 Liu Y, Zhang GA, Yu PN, *et al*. BioCopy: A plug-and-play span copy mechanism in Seq2Seq models. Proceedings of the 2nd Workshop on Simple and Efficient Natural Language Processing. Association for Computational Linguistics, 2021. 53–57.

25 Li HR, Xu S, Yuan P, *et al*. Learn to copy from the copying history: Correlational copy network for abstractive summarization. Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Punta Cana: Association for Computational Linguistics, 2021. 4091–4101.

26 Xiao W, Carenini G. Entity-based SpanCopy for abstractive summarization to improve the factual consistency. Proceedings of the 4th Workshop on Computational Approaches to Discourse. Toronto: Association for Computational Linguistics, 2023. 70–81.

27 Lin CY, Hovy E. Automatic evaluation of summaries using n-gram co-occurrence statistics. Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics. Edmonton: The Association for Computational Linguistics, 2003. 150–157.

(校对责编: 孙君艳)