北京大学学报(自然科学版) 第 56 卷 第 1 期 2020 年 1 月
Acta Scientiarum Naturalium Universitatis Pekinensis, Vol. 56, No. 1 (Jan. 2020)
doi: 10.13209/j.0479-8023.2019.100

基于编码器共享和门控网络的生成式文本摘要方法

田珂珂[1,2] 周瑞莹[1,2] 董浩业[1,2] 印鉴[1,2,†]

1. 中山大学数据科学与计算机学院, 广州 510006; 2. 广东省大数据分析与处理重点实验室, 广州 510006;
† 通信作者, E-mail: issjyin@mail.sysu.edu.cn

摘要 结合基于自注意力机制的 Transformer 模型, 提出一种基于编码器共享和门控网络的文本摘要方法。该方法将编码器作为解码器的一部分, 使解码器的部分模块共享编码器的参数, 同时使用门控网络筛选输入序列中的关键信息。相对已有方法, 所提方法提升了文本摘要任务的训练和推理速度, 同时提升了生成摘要的准确性和流畅性。在英文数据集 Gigaword 和 DUC2004 上的实验表明, 所提方法在时间效率和生成摘要质量上, 明显优于已有模型。

关键词 生成式; 文本摘要; 自注意力机制; 编码器共享; 门控网络

An Abstractive Summarization Method Based on Encoder-Sharing and Gated Network

TIAN Keke[1,2], ZHOU Ruiying[1,2], DONG Haoye[1,2], YIN Jian[1,2,†]

1. School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006; 2. Guangdong Key Laboratory of Big Data Analysis and Processing, Guangzhou 510006; † Corresponding author, E-mail: issjyin@mail.sysu.edu.cn

Abstract This paper proposed an abstractive summarization method based on self-attention based Transformer model, which regarded encoder as part of decoder, and used gated network to control the information flow from encoder to decoder. Compared with the existing methods, proposed method improves the training and inference speed of text summarization task, and improves the accuracy and fluency of generating summary. Experiments on English summarization dataset Gigaword and DUC2004 demonstrate that proposed model outperforms the baseline models on both the quality of summarization and time efficiency.

Key words abstractive; summarization; self-attention; encoder-sharing; gated network

自动文本摘要旨在对给定的一段长文本进行压缩、精简, 并产生一段简洁、流畅且保留原文关键信息的短文本。文本摘要的意义在于缓解互联网时代人们面临的信息过载问题, 通过对文本进行压缩, 提取其主要信息, 可以大大降低用户的阅读成本, 帮助用户更高效地从互联网获取所需信息。

目前文本摘要方法可分为两大类: 抽取式方法(extractive)和生成式方法(abstractive)。图 1 展示两种方法生成的摘要。抽取式方法是按照一定的规则, 在原文中抽取句子、短语和词组成摘要。该方法产生的摘要通常较为冗长, 且多个摘要句之间可能产生语义不连贯的现象。生成式方法是通过阅读原文内容提取关键信息, 并重新组织文字生成摘要, 与人工做摘要的方式相似, 生成的摘要也较简洁, 近年来得到广泛应用。

本文提出的方法属于生成式方法。生成式方法遵循编码–解码框架, 编码器用于阅读原文, 并提取主要信息, 解码器根据编码器提取的信息生成摘

广东省科技计划项目(2015A030401057, 2016B030307002, 2017B030308007)资助
收稿日期: 2019–05–22; 修回日期: 2019–09–25

“#”表示数字, 所有的词都已小写化

图 1 抽取式方法和生成式方法生成的摘要对比

Fig. 1 Summaries generated by extractive and abstractive method

要。以往的生成式方法通常使用循环神经网络作为编码器和解码器, 这些方法在文本摘要领域取得很好的效果。但是, 循环神经网络的结构特点——逐词处理序列, 使其难以并行化, 无论在训练阶段还是测试阶段, 效率都比较低, 当序列较长时, 这个问题尤为突出。

Vaswani等[1]提出完全基于注意力机制的Transformer模型, 不使用循环神经网络, 可以减少训练时间, 并刷新机器翻译任务的表现。鉴于Transformer良好的并行能力, 本文基于Vaswani等[1]提出的编码器共享和门控网络的Transformer, 在解码器与编码器之间进行参数共享, 减少模型的参数, 强化对编码器的训练, 并使用多层感知机作为门控网络, 用以控制从编码器到解码器的信息流, 仅传递关键信息, 使得模型更关注原文中的重要信息, 以便生成更准确精简的摘要。

1 相关工作

文本摘要任务在形式上与机器翻译任务相似, 输入为一个序列, 输出也是一个序列。近年来, 许多机器翻译领域的方法被应用到文本摘要任务中。Sutskever等[2]提出seq2seq模型, 包括编码器和解码器两部分。编码器将输入序列映射到固定长度的向量上, 解码器根据该向量解码得到输出序列, 该模型用于解决英–法翻译问题, 取得巨大成功。Bahdanau等[3]提出注意力机制, 使得解码器在产生输出序列时, 不只是利用一个固定长度的向量, 而是可以回看输入序列的信息, 大大提升机器翻译的效果。此后, 注意力机制成为处理所有序列到序列问题(如机器翻译、文本摘要和语音识别等)时必不可少的一个模块。

Rush等[4]提出第一个生成式文本摘要方法, 使用带注意力机制的卷积神经网络作为编码器, 神经网络语言模型作为解码器, 并第一次使用Gigaword数据集和DUC2004数据集完成文本摘要任务。Hu等[5]提出一个新的中文文本摘要数据集LCSTS来填补中文文本摘要数据上的空缺, 推动国内文本摘要领域的发展。Chopra等[6]在文献[4]的基础上进行改进, 使用循环神经网络作为解码器, 编码器仍使用卷积神经网络, 提升了生成摘要的质量。Nallapati等[7]提出完全基于循环神经网络的seq2seq模型, 编码器和解码器都使用循环神经网络, 同时引入一些词汇特征(如词性和命名实体等), 进一步提升模型表现。针对未登录词问题(部分低频词不在词表中, 无法编码, 也无法生成为摘要序列的一部分), Gu等[8]提出拷贝机制(copy mechanism), 使得模型在生成摘要时, 可以选择从输入序列复制一段话, 而不仅仅是从词汇表生成词语, 解决了上述问题。Zhou等[9]提出选择性编码模型, 用于对输入序列的词进行筛选, 只保留关键信息, 从而实现对输入序列的选择性编码。Lin等[10]提出全局编码模型, 使用卷积门控网络对输入序列进行筛选, 在文本摘要任务上达到领先水平。Paulus等[11]将强化学习引入文本摘要中, 直接针对摘要的评分指标进行优化, 减轻了曝光偏差问题, 进一步提升摘要表现。Vaswani等[1]提出新的序列到序列模型Transformer, 既不使用循环神经网络, 也不使用卷积神经网络, 而是完全依赖于注意力机制, 在序列自身各个词之间计算注意力权重, 得到每个词的上下文表示, 因此又称为自注意力机制。该模型的训练时间远远少于此前的序列到序列模型, 并刷新了机器翻译任务的BLEU得分。

2 基于编码器共享和门控网络的生成式文本摘要方法

本文基于Vaswani[1]等提出的Transformer模型进行改进, 图2展示本文模型, 包含编码器、门控

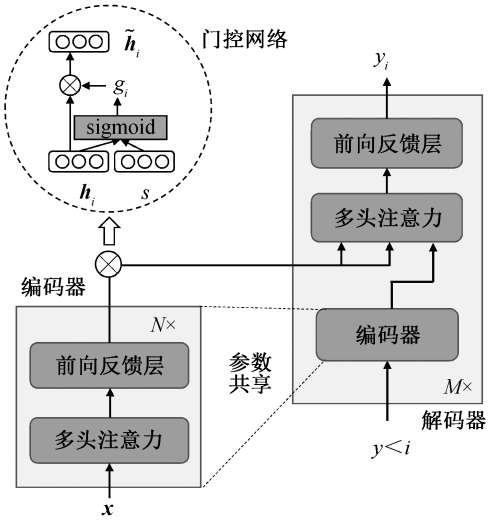

图 2　本文模型概览

Fig. 2　An overview of proposed model

网络和解码器 3 个部分。编码器用于读取输入序列 $\boldsymbol{x}=\{x_0, x_1, \ldots, x_n\}$，并产生该序列对应的向量表示 $\boldsymbol{h}=(h_0, h_1, \ldots, h_n)$；门控网络用于对编码器的输出 h 进行筛选，去除无用信息，即对每个向量表示 h_i 产生一个实数值 $g_i \in [0, 1]$，进一步得到 $\tilde{\boldsymbol{h}}=(g_0h_0, g_1h_1, \ldots, g_nh_n)$，以达到筛选的目的；解码器根据 $\tilde{h}$ 来产生摘要序列。

2.1 问题形式化

给定一段输入序列 $\boldsymbol{x}=\{x_1, x_2, \ldots, x_n\}$，其中 n 表示序列长度。文本摘要系统的目标是输出一段摘要序列 $\boldsymbol{y}=\{y_1, y_2, \ldots, y_m\}$，其中 $m(m \leqslant n)$为摘要序列长度。在训练阶段，我们训练模型，使其生成的摘要 $\boldsymbol{y}$ 尽量与参考摘要 $\hat{\boldsymbol{y}}$ 相同；在测试阶段，模型根据输入序列 $\boldsymbol{x}$ 来生成摘要序列。

2.2 编码器

编码器的作用是读取输入序列，并对每个词产生一个向量表示。为了高效地对输入序列进行编码，我们使用基于自注意力机制的编码器，相对于循环神经网络，不需要逐词处理输入序列，而是通过自注意力机制同时计算每个词的上下文向量，因此有良好的并行能力，计算复杂度较低。

如图 2 所示，编码器可堆叠 N 层，每层包括多头注意力层和前向反馈层。多头注意力层用于在输入序列内计算每个词关于其他词的注意力权重，以便得到每个词的上下文表示，“多头”的意思是将输入映射到多个子空间，并在这些子空间内计算上下文表示，最后将计算结果拼接在一起，如式(1)所示。

$$\text{Multihead}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)\boldsymbol{W}^O, \tag{1}$$

其中，$\text{head}_i = \text{Attention}(\boldsymbol{QW}_i^Q, \boldsymbol{KW}_i^K, \boldsymbol{VW}_i^V)$，$\boldsymbol{Q}=\boldsymbol{K}=\boldsymbol{V}=\boldsymbol{x}$；参数矩阵 $\boldsymbol{W}_i^Q \in R^{d\times d_k}$，$\boldsymbol{W}_i^K \in R^{d\times d_k}$，$\boldsymbol{W}_i^V \in R^{d\times d_v}$，$\boldsymbol{W}^O \in R^{hd_v\times d}$ 表示线性转换，用于将输入映射到不同的子空间；d_k 和 d_v 表示子空间的维度；h 表示多头注意力层的头数；d 表示模型隐藏层大小。注意力函数为放缩点积注意力函数，如式(2)所示：

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{QK}^{\mathrm{T}}}{\sqrt{d}}\right)\boldsymbol{V}。 \tag{2}$$

前向反馈层(feed forward networks, FFN)作用于多头注意力层的输出，包含两个线性转换操作和 ReLU 激活函数[12]，用于增加模型的非线性拟合能力，如式(3)所示：

$$\text{FFN}(\boldsymbol{x}) = \max(0, \boldsymbol{x}\boldsymbol{W}_1 + b_1)\boldsymbol{W}_2 + b_2, \tag{3}$$

其中 $\boldsymbol{W}_1 \in R^{d\times d_{\text{ff}}}$，$\boldsymbol{W}_2 \in R^{d_{\text{ff}}\times d}$ 为线性转换；d_{ff} 为该层的隐藏层大小，b_1 和 b_2 为偏置。

本文使用的参数为 h=4，d=256，d_{ff}=1024，$d_k=d_v=d/h$=64，编码器层数 N=2。值得注意的是，本文模型的编码器不仅负责对输入序列进行编码，也会作为解码器的一部分，对摘要序列进行编码。

2.3 门控网络

输入序列中通常包含许多词，其中只有少部分词包含整个序列的关键信息，这些关键信息也正是模型在生成摘要时所需要的。为了使模型能对输入序列的关键信息进行筛选，我们提出如图 2 所示的门控网络。

门控网络用于控制从输入序列到输出序列的信息流，去除无用信息，使解码器能更专注于从关键信息中生成摘要。门控网络的输入为原文序列的句子表示 s 以及该序列中某个词的词表示 h_i；输出为对 h_i 进行筛选得到的新向量表示 $\tilde{h}_i$。参考 Devlin 等[13]的工作，本文也将 h_0(输入序列的起始标识符对应的隐藏层表示)作为输入序列向量的表示，即 $s=h_0$。对于每个词表示 h_i，门控网络都会生成一个阈值 g_i：

$$g_i = \text{sigmoid}(\boldsymbol{W}_g[h_i, s] + b), \tag{4}$$

其中，$\boldsymbol{W}_g \in R^{2d\times 1}$ 为线性转换，b 表示偏置。g_i 越大，表示该词越关键。通过 g_i 来控制 h_i 通往解码器的信

息量，得到筛选后的向量 $\widetilde{h}_i$，如式(5)所示:

$$\widetilde{h}_i = g_i h_i。 \tag{5}$$

对每个词进行筛选后，得到整个序列的向量表示 $(\widetilde{h}_0, \widetilde{h}_1, \ldots, \widetilde{h}_n)$。然后，此向量序列将传递给解码器，用于生成摘要。

2.4 共享编码器参数的解码器

解码器根据门控网络的输出信息生成摘要序列。首先，读取已经产生的摘要序列 $\boldsymbol{y}_{<i}=\{y_0, y_1, \ldots, y_{i-1}\}$(开始时摘要序列仅包含开始标识符，如“<s>”)，并对其进行编码，进而产生向量序列 $\boldsymbol{s}_{<i}=\{s_0, s_1, \ldots, s_i\}$。然后根据 $\boldsymbol{s}_{<i}$ 和门控网络的输出 $\tilde{\boldsymbol{h}}$，预测摘要序列的下一个词 y_i。依此类推，最终得到摘要序列。

解码器的功能之一是对已经产生的摘要序列进行编码，得到其向量表示。这一点与编码器的功能相似，不同之处在于，编码器是对输入序列进行编码，而解码器该是对摘要序列进行编码，但这并不影响两个模块功能上的相似性。基于这种相似性，我们提出编码器共享(encoder-sharing)的解码器，将编码器作为解码器的模块之一，把已生成序列 $y_{<i}$ 的编码任务交给编码器。与文献[1]相比，本文方法的优势如下。

1) 整合了冗余的功能模块，减少了模型参数，降低模型复杂度。文献[1]使用一个额外的多头注意力层对摘要序列进行编码，本文去掉这个模块，将编码任务交给编码器。

2) 文献[1]中，编码器的训练数据只有输入序列，本文还包括摘要序列，通过更多数据的训练，能够增强编码器的编码能力。

3) 通过使用同一个编码器，将输入序列和摘要序列映射到同一向量空间。解码器需要在两个序列之间计算注意力权重，两个向量表示处于同一个向量空间，更有利于点积注意力函数的计算，能更清楚地挖掘输入序列与输出序列之间的关系。

解码器还包含另外两个模块：多头注意力层和前向反馈层。多头注意力层用于在输入序列和摘要序列之间计算注意力权重(式(1))，与编码器中的多头注意力层不同的是 $\boldsymbol{Q}=\boldsymbol{s}_{<i}$，$\boldsymbol{K}=\boldsymbol{V}=\tilde{\boldsymbol{h}}$。前向反馈层用于增加模型的非线性拟合能力(式(2))。经过这两个层之后，得到输出向量 m_i，最终经过 softmax 层预测下一个词，如式(6)所示:

$$p(y_i|\boldsymbol{x}, \boldsymbol{y}_{<i}) = \mathrm{softmax}(\boldsymbol{m}_i \boldsymbol{W}_o + b_o), \tag{6}$$

其中，$W_o \in R^{d\times|V|}$，$|V|$ 表示词汇表大小，$b_o \in R^{|V|}$ 表示偏置。解码器可堆叠 M 层，本文中 M=2，其中多头注意力层和前向反馈层的参数配置与编码器相同。

2.5 目标函数

在训练阶段，模型的目标是在给定输入序列 x 后，最大化地得到摘要序列 y 的概率，因此目标函数是最小化的负对数似然函数:

$$J(\theta) = -\frac{1}{|D|}\sum_{(x,y)\in D} \log p(\boldsymbol{y}|\boldsymbol{x}), \tag{7}$$

其中，D 代表训练集，θ 表示模型参数集合，$p(\boldsymbol{y}|\boldsymbol{x})$ 表示给定输入序列 $\boldsymbol{x}$ 的情况下，得到摘要序列 $\boldsymbol{y}$ 的概率。

3 实验与结果分析

3.1 数据集

本文使用 Gigaword 数据集和 DUC2004 数据集，详细情况如表 1 所示。对于 Gigaword 数据集，我们使用 Rush[4]等预处理后的版本，处理过程如下：1) 所有英文单词都小写化；2) 数字用“#”代替；3) 出现次数少于 5 次的单词用未登录词标识符“<unk>”代替。Gigaword 数据集较大，可划分为训练集、验证集和测试集 3 部分，分别包含约 380 万、18 万和 1951 个文本–摘要对，用于模型训练和测试。DUC 2004 数据集仅包含 500 篇文档，每篇文档对应 4 条参考摘要，因该数据集较小，不适合模型训练，所以使用已在 Gigaword 数据集上训练好的模型，在 DUC2004 数据集上进行测试。

3.2 评估指标

我们使用 ROUGE 指标[14]评估生成摘要的质量。ROUGE 指标主要评估模型生成摘要与标准摘要之间的重合度，重合度越高，ROUGE 得分也越高。ROUGE 从 3 个粒度来评判重合度：词、二元短语和最长公共子序列，对应 3 个指标：ROUGE-1, ROUGE-2 和 ROUGE-L，每个指标都包含精确率、召回率和 F1 值。F1 值用来评判模型在 Gigaword 数据集上的效果，召回率用来评判模型在 DUC2004

表 1 Gigaword 和 DUC2004 数据集的详细情况

Table 1 Details about Gigaword and DUC2004 dataset

数据集	句子数	参考摘要数量	平均输入序列长度	平均摘要长度
Gigaword	3.99×10^6	1	31.4	8.2
DUC2004	500	4	35.6	10.3

数据集上的效果。

3.3 实现过程

我们基于Gigaword训练集的源文本和摘要文本建立源词典和目标词典，其大小分别为90000和68883；词向量维度大小为256；编码器和解码器的层数都设置为2，所有的多头注意力层都包含4个头，隐藏层神经元个数为256；前向反馈层的中间层大小为1024；使用位置编码[1]，以便利用序列的顺序信息，各个网络层之间采用残差连接[15]，避免梯度消失；使用dropout方法[16]来避免过拟合，比率设置为0.1。

在训练阶段，随机打乱训练集，采用批量训练，批大小设置为64；使用Adam优化器[17]，初始学习率设为0.001，在训练集上每训练两次后，将学习率衰减至当前的一半。训练10个epoch后，模型趋于收敛。使用一块Nvidia GTX 1080ti显卡进行训练。

在测试阶段，使用束搜索方法来选择候选摘要序列，束宽度设置为5。束搜索倾向于选择较短的摘要句，为了避免此缺点，我们将搜索过程中各个候选摘要的得分除以其长度，用以鼓励模型生成更长的摘要。

3.4 模型说明

本文提出的两个模型如下。

ES-Trans：编码器共享的Transformer，在文献[1]的基础上进行改进，将编码器作为解码器的一部分。

Gated-ES-Trans：在ES-Trans的基础上，引入门控网络，用以控制从编码器到解码器的信息流。

我们将本文模型与以下基线模型进行对比。

ABS[4]：使用卷积神经网络作为编码器，神经网络语言模型NNLM作为解码器，首次在Gigaword数据集上评估其文本摘要方法。

RAS[6]：使用卷积神经网络作为编码器，循环神经网络作为解码器。

Feats2s[7]：编码器和解码器都使用循环神经网络，是第一个完全基于循环神经网络的模型，还使用了多个特征，如词性和实体信息等。

Entail-s2s[18]：基于循环神经网络，引入门控网络和文本蕴涵任务来进行多任务训练，增强编码器提取关键信息的能力。

Seq2seq：使用双向门控循环单元(gated recurrent unit，GRU)[19]作为编码器，单向GRU作为解码器，使用点积注意力函数。本文编写代码，并在两个数据集上进行实现。

Transformer[1]：既不使用循环神经网络，也不使用卷积神经网络，而是完全依赖于注意力机制。本文编写代码，并在数据集上实验。

3.5 结果与分析

我们在英文摘要数据集Gigaword和DUC2004上进行测试，并报告各个模型对应的ROUGE-1，ROUGE-2和ROUGE-L得分，使用封装了ROUGE脚本的pyrouge工具(https://pypi.org/project/pyrouge)来计算得分。此外，我们列出每个基线模型在训练集上训练一次的时间以及推理速度(推理阶段采用束搜索方法，束宽度设置为8)，用来对此模型的时间效率。

表2展示各个模型在Gigaword数据集上的ROUGE指标的F1评分以及训练时间。本文模型Gated ES-Trans在ROUGE-1、ROUGE-2和ROUGE-L指标上的得分都明显优于其他模型，与已有的最优模型Entail-s2s相比，得分分别提升0.82，0.24和

表2　在Gigaword数据集上各模型的ROUGE F1评分以及训练和测试时间

Table 2　ROUGE F1 score and training and testing cost of each model on Gigaword dataset

模型	F1/%			训练时间 ($h\cdot epoch^{-1}$)	推理速度 ($items\cdot s^{-1}$)
	ROUGE-1	ROUGE-2	ROUGE-L		
ABS	29.55	11.32	26.42	4.2	6.4
Feats2s	32.67	15.59	30.64	9.8	2.3
RAS	33.78	15.97	31.15	5.1	6.6
Entail-s2s	35.33	17.27	33.19	4.7	7.1
Seq2seq	33.20	15.64	30.82	3.5	8.3
Transformer	33.08	15.37	30.49	1.3	18.3
ES-Trans (本文)	35.98	17.30	33.24	1.4	18.1
Gated ES-Trans (本文)	36.15	17.51	33.47	1.4	17.2

0.28; 比RAS模型得分提升分别超过 2.0, 1.5和2.0, 同时训练时间仅为其1/3甚至更少。Transformer模型相比, 本文模型ES-Trans大幅度提升ROUGE得分, 分别达到2.90, 1.93和2.75。此外, 相对于已有的生成式方法, 本文方法大大缩短训练时间, 提升了推理速度, 加入门控网络后的Gated ES-Trans模型, 取得更高的ROUGE 得分, 说明了门控网络的有效性。

表 3 展示各个模型在 DUC2004 测试集上的 ROUGE 召回率, 本文使用在 Gigaword 数据集上训练好的模型, 直接在 DUC2004 数据集上进行测试。可见, 本文方法 Gated ES-Trans 比 RAS 和 Transformer 的 ROUGE-2 和 ROUGE-L 有明显提升, 与 Entail-s2s 的效果相当, ROUGE-1 和 ROUGE-2 评分稍低, 但 ROUGE-L 评分更高。

表 3 在 DUC2004 测试集上各模型的 ROUGE 召回率评分(%)

Table 3 ROUGE recall of each model on DUC2004 dataset (%)

模型	ROUGE-1	ROUGE-2	ROUGE-L
ABS	26.55	7.06	22.05
Feats2s	28.35	9.46	24.59
RAS	28.97	8.26	24.06
Entail-s2s	**29.33**	**10.24**	25.24
Seq2seq	27.88	8.41	24.04
Transformer	27.54	8.29	24.01
ES-Trans (本文)	29.02	9.81	25.12
Gated ES-Trans (本文)	29.25	9.97	**25.41**

说明: 加粗数字表示最优结果。

3.6 案例分析

本节从 Gigaword 测试集中摘取两个例子, 展示本文最终模型与 Vaswani 等[1]的 Transformer 模型在两个示例上的摘要结果, 对比两个方法生成摘要的正确性和完整性, 如表 4 所示。

例 1 的主要内容为“斯里兰卡因战争动荡关闭学校”, 本文模型产生的摘要基本上完整地概括了这一内容, 与参考摘要相差无几。Transformer模型产生的摘要明显地提取错重点, 其生成摘要内容“斯里兰卡发起对叛军的军事行动”虽然在原文中有所体现, 但并不是原文所要表达的主要意思。第二个示例描述了有关尼亚加拉飞机失事以及人员伤亡情况。Trans-former 模型产生的摘要, 正确地说明了人员伤亡情况, 但未说明原因, 存在信息遗漏; 本文模型完整地概括了伤亡情况及原因。两个例子证明了本文模型生成摘要的正确性及完整性。

4 结语

本文提出基于编码器共享和门控网络的生成式文本摘要方法, 将编码器作为解码器的一部分, 使编码器不只对输入序列进行编码, 也能对已产生的摘要序列进行编码, 进而增强对编码器的训练; 同时引入门控网络, 对输入序列的信息进行筛选, 保留关键信息, 使得模型能根据原文的关键信息来生成摘要。在英文摘要数据集 Gigaword 和 DUC 2004 上的实验证明, 本文提出的基于编码器共享和门控网络的方法能明显提升模型的训练速度和生成摘要的质量。

表 4 Gigaword 数据集中的两个示例以及模型对应的摘要

Table 4 Two examples from Gigaword and corresponding summarization

例 1	输入	the sri lankan government on wednesday announced the closure of government schools with immediate effect as a military campaign against tamil separatists escalated in the north of the country .
	参考摘要	sri lanka closes schools as war escalates
	Transformer 摘要	sri lanka launches campaign against tamil rebels
	Gated ES-Trans 摘要	sri lanka closes schools as military campaign escalates
例 2	输入	at least ## people were killed when a nigeria airways airliner crashed on landing monday at kaduna airport in the north of the country , airport officials said.
	参考摘要	nigerian plane crashes on landing killing at least ##
	Transformer 摘要	at least ## killed in nigeria
	Gated ES-Trans 摘要	at least ## killed in nigerian airways plane crash

参考文献

[1] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need // 31st Conference on Neural Information Processing Systems. Long Beach: MIT Press, 2017: 6000–6010

[2] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks // 28th Conference on Neural Information Processing Systems. Montreal: MIT Press, 2014: 3104–3112

[3] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translation [EB/OL]. (2016–05–19) [2019–05–20]. https://arxiv.org/abs/1409.0473

[4] Rush A M, 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: ACL Press, 2015: 379–389

[5] Hu Baotian, Chen Qingcai, Zhu Fangze. LCSTS: a large scale chinese short text summarization dataset // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. Lisbon: ACL Press, 2015: 1967–1972

[6] Chopra S, Auli M, Rush A M. Abstractive sentence summarization with attentive recurrent neural networks // The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego: ACL Press, 2016: 93–98

[7] Nallapati R, Zhou B, Santos C, et al. Abstractive text summarization using sequence-to-sequence RNNs and beyond // Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning. Berlin: ACL Press, 2016: 280–290

[8] Gu Jiatao, Lu Zhengdong, Li Hang, et al. Incorporating copying mechanism in sequence-to-sequence learning // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL Press, 2016: 1631–1640

[9] Zhou Qingyu, Yang Nan, Wei Furu, et al. Selective encoding for abstractive sentence summarization // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: ACL Press, 2017: 1095–1104

[10] Lin Junyang, Sun Xu, Ma Shuming, et al. Global encoding for abstractive summarization // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: ACL Press, 2018: 163–169

[11] Paulus R, Xiong Caiming, Socher R. A deep reinforced model for abstractive summarization [EB/OL]. (2017–11–13)[2019–05–20]. https://arxiv.org/abs/1705.04304

[12] Nair V, Hinton G E. Rectified linear units improve restricted Boltzmann machines // Proceedings of the 27th International Conference on Machine Learning. Haifa: Omnipress, 2010: 807–814

[13] Devlin J, Chang M, Lee K. BERT: pre-training of deep bidirectional transformers for language understanding [EB/OL]. (2019–05–24) [2019–05–30]. https://arxiv.org/abs/1810.04805

[14] Lin C. Rouge: a package for automatic evaluation of summaries // Proceedings of the ACL Workshop: Text Summarization Braches Out. Barcelona, 2004: 74–81

[15] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition // 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016: 770–778

[16] Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research. 2014, 15(1): 1929–1958

[17] Kingma D P, Ba J. Adam: a method for stochastic optimization [EB/OL]. (2017–01–30) [2019–05–20]. https://arxiv.org/abs/1412.6980

[18] Li Haoran, Zhu Junnan, Zhang Jiajun, et al. Ensure the correctness of the summary: incorporate entailment knowledge into abstractive sentence summarization // COLING 2018. Santa Fe, 2018: 1430–1441

[19] Cho K, Merrienboer B, Gulcehre C, et al. Learning phrase representations using rnn encoder-decoder for statistical machine translateion // Proceddings of the 2014 conference on Empirical Methods in Natural Language Processing. Doha: ACL Press, 2014: 1724–1734