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 广东 广州 510006;

† 通讯作者，E-mail: issjyin@mail.sysu.edu.cn

摘要　以往的生成式文本摘要方法通常使用循环神经网络来作为其编码器或解码器，在生成的摘要质量方面取得了较好的表现，但限于循环神经网络逐词处理序列的特点，其训练和推理的时间效率较为低下。针对这个问题，结合基于自注意力机制的 Transformer 模型，提出了基于编码器共享和门控网络的文本摘要方法，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，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 的实验表明，所提出的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；循环神经网络

中图分类号

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, P.R.China;

†Corresponding Author, E-mail: issjyin@mail.sysu.edu.cn

Abstract　Previous abstractive text summarization methods use Recurrent Neural Networks as encoder and decoder, which have an intrinsic deficiency in training speed and inference speed, since they process a sequence word by word. To address this problem, we proposed a self-attention based abstractive summarization method, which regards encoder as part of decoder, and use gated network to control the information flow from encoder to decoder. Experiments on the English summarization dataset Gigaword and DUC2004 demonstrate that our model outperforms the baseline models on both the quality of summarization and time efficiency.

Key words　abstractive, summarization, self attention, encoder-sharing, recurrent neural networks

1　引言

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。文本摘要出现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，大大降低人们的阅读成本，帮助用户更高效的从互联网中获取所需要的信息。

已有的文本摘要方法可分为两大类，分别是抽取式方法（Extractive）和生成式方法（Abstractive）。抽取式方法通过在原文中按照一定规则抽取句子、短语、词来组成摘要，该类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现象；生成式方法通过阅读原文内容，提取其关键信息，并重新组织文字来生成摘要，跟人类做摘要的方式十分相似，生成的摘要也较简洁，故生成式方法在近年来得到了广泛应用，本文所提出的方法属于生成式方法。图 1 展示了抽取式方法和生成式方法生成的摘要。

本文研究得到国家自然科学基金(U1611264) 、广东省和广州市科技计划项目(2014B030301007, 2015A030401057,2016B030307002, 2014SY000013, 2017B030308007) 资助

生成式方法遵循编码-解码框架，编码器用于阅读原文，并提取其主要信息，解码器用于根据编码器提取的信息来生成摘要。以往的方法通常使用循环神经网络来作为编码器和解码器，这些方法在文本摘要领域中取得了很好的效果。但循环神经网络本身的结构特点——逐词处理序列，使其难以被并行化，无论在训练阶段还是测试阶段，时间效率都十分低下，序列较长的情况下这个问题尤为突出。2017 年，谷歌的Vaswani[1]等提出了完全基于注意力机制的 Transformer 模型，不使用循环神经网络，大大减少了训练时间，并且刷新了机器翻译任务的表现。鉴于 Transformer 良好的并行能力，本文基于其提出编码器共享和门控网络的 Transformer (Gated Encoder-sharing Transformer, Gated ES-Trans)，在解码器和编码器之间进行参数共享，减少了模型的参数，强化了对编码器的训练；并使用了多层感知机作为门控网络，以控制从编码器到解码器的信息流，仅传递关键信息，使得模型更能关注到原文中的重要信息，以生成更准确精简的摘要。在英文摘要数据集 Gigaword 和 DUC2004 上的实验证明，相对以往方法，本文方法大大提升了时间效率，同时在生成的摘要质量方面，也明显优于已有方法。

图 1　抽取式方法和生成式方法生成的摘要对比（“#”表示数字，所有的词都已小写化）

Fig.1　The summaries generated by extractive and abstractive method (in lower case, “#” stand for digit)

2　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。故近年来，许多机器翻译领域的方法都被应用到文本摘要任务上。2014 年，Sutskever[2]等提出 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，而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，并获得了巨大的成功。同年，Bahdanau[3]等提出了注意力机制，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，大大提升了机器翻译的效果，此后，注意力机制也成为所有序列到序列问题（如机器翻译、文本摘要、语音识别等）必不可少的一个模块。

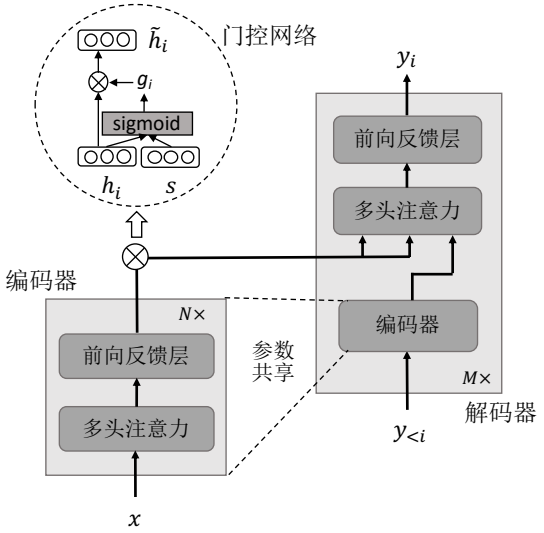
2015 年，Rush[4]等提出了第一个生成式文本摘要方法，使用带注意力机制的卷积神经网络来作为编码器，神经网络语言模型作为解码器，并第一次使用 Gigaword 数据集和 DUC2004 数据集进行文本摘要任务，这一工作开启了生成式文本摘要的篇章。同年，Hu[5]等人提出一个新的中文文本摘要数据集 LCSTS，填补了中文文本摘要数据上的空缺，推动了国内文本摘要领域的发展。2016 年，Chopra[6]等在 Rush 工作的基础上进行改进，使用循环神经网络作为解码器，而编码器仍使用卷积神经网络，提升了生成摘要的质量。同年，Nallapati[7]等提出完全基于循环神经网络的 seq2seq 模型，编码器和解码器都使用循环神经网络，同时引入了一些词汇特征（如词性，命名实体等），进一步提升了模型表现。针对未登录词问题（部分低频词不在词表中，无法被编码，也无法被生成为摘要序列的一部分），Gu[8]等提出了拷贝机制（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，而不只是从词汇表生成词语，进而解决了上述问题。2017 年，Zhou[9]等提出了选择性编码模型，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[10]等提出全局编码模型，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[11]等将强化学习引入到文本摘要中，直接针对摘要的评分指标进行优化，减轻了曝光偏差问题，进一步提升了摘要表现。2017 年，Vaswani[1]等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，以得到每个词的上下文表示，因此又叫做自注意力机制。该模型的训练时间远远少于已有序列到序列模型，并且刷新了机器翻译任务的 BLEU 得分。

3 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 展示了本文模型，包含编码器、门控网络、解码器三部分。编码器用于读取输入序列$x = \{x_0, x_1, \dots, x_n\}$，并产生该序列对应的向量表示$h = (h_0, h_1, \dots, h_n)$；门控网络用于对编码器的输出$h$进行筛选，去除无用信息，具体来说，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$\tilde{h} = (g_0 h_0, g_1 h_1, \dots, g_n h_n)$，以达到筛选的目的；最后解码器根据$\tilde{h}$来产生摘要序列。接下来我们详细介绍编码器、门控网络、解码器。

3.1 问题形式化

给定一段输入序列$x = \{x_1, x_2, \dots, x_n\}$，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$y = \{y_1, y_2, \dots, y_m\}$，其中$m(m \le n)$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y尽量与参考摘要$\hat{y}$相同，测试阶段模型根据输入序列x来生成摘要序列。



本文模型包含编码器、门控网络、解码器。编码器用于读取输入序列 x，并提取其语义向量；门控网络对编码器的输出进行筛选，保留关键信息；解码器根据已生成的摘要序列$y_{<i}$和门控网络的输出，来生成摘要序列的下一个词y_i，依此类推，最终生成整个摘要序列。

图 2　本文模型概览

Fig.2　An overview of our proposed model

3.2 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。为了能高效的对输入序列进行编码，我们使用了基于自注意力机制的编码器，其相对循环神经网络，不需要逐词处理输入序列，而是通过自注意力机制来同时计算每个词的上下文向量，因此有着良好的并行能力，计算复杂度较低。

如图 2 所示，编码器可堆叠多层，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它词的注意力权重，以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，如公式（1）所示。

$$Multihead(Q,K,V) = concat(head_1, head_2, \dots, head_h)W^O \tag{1}$$

$$\text{其中 } head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

其中$Q = K = V = x$，参数矩阵$W_i^Q \in R^{d\times d_k}$，$W_i^K \in R^{d\times d_k}$，$W_i^V \in R^{d\times d_v}$，$W^O \in R^{hd_v\times d}$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击注意力函数，如公式（2）所示：

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

前向反馈层（Feed Forward Networks，FFN）作用于多头注意力层的输出，包含了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，如公式（3）所示。

$$FFN(x) = max(0, xW_1 + b_1)W_2 + b_2 \tag{3}$$

其中$W_1 \in R^{d \times d_{ff}}$，$W_2 \in R^{d_{ff} \times d}$为线性转换，$d_{ff}$为该层的隐藏层大小，$b_1$和$b_2$为偏置。

在本文中，我们使用的参数为$h = 4$，$d = 256$，$d_{ff} = 1024$，$d_k = d_v = d/h = 64$，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，以对摘要序列进行编码，在后续关于解码器的介绍中会进行详细阐述。

3.3 门控网络

输入序列中通常包含许多词，而其中仅有少部分词，包含了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，我们提出了如图 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，从而使解码器能更专注于从关键信息中生成摘要。具体来说，门控网络结构如图 2 所示，其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[13]等的工作，其将h_0（输入序列的起始标识符对应的隐藏层表示）作为句子的向量表示，故在我们的工作中，也将h_0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，如公式（4）：

$$g_i = sigmoid(W_g[h_i, s] + b) \tag{4}$$

其中$W_g \in R^{2d \times 1}$为线性转换，b 表示偏置，g_i越大，表示该词越关键。接着通过g_i来控制h_i通往解码器的信息量，以得到筛选后的向量$\tilde{h}_i$，如公式（5）所示。对每个词进行筛选后得到整个序列的向量表示$(\tilde{h}_0, \tilde{h}_1, \dots, \tilde{h}_n)$。而后，此向量序列将传递给解码器，用于生成摘要。

$$\tilde{h}_i = g_i h_i \tag{5}$$

3.4 共享编码器参数的解码器

解码器用于根据门控网络的输出信息，来生成摘要序列。具体来说，解码器首先读取已经产生的摘要序列$y_{<i} = \{y_0, y_1, \dots, y_{i-1}\}$（最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$s_{<i} = \{s_0, s_1, \dots, s_i\}$。接着根据$s_{<i}$和门控网络的输出$\tilde{h}$，来预测下摘要序列的下一个词$y_i$，依此类推，最终得到摘要序列。

可见解码器的功能之一，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，不同点在于，编码器是对输入序列进行编码，而解码器该部分功能是对摘要序列进行编码，但这一不同点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，将对已生成序列$y_{<i}$的编码任务交给了编码器。相对于 Vaswani[1]等的工作，这样做的好处如下：

（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。原来 Vaswani 等的工作，使用一个额外的多头注意力层，来对摘要序列进行编码，而我们去掉这个模块，将编码任务交给了编码器。

（2）相当于增加了编码器的训练数据，原来编码器的训练数据只有输入序列，现在也包括了摘要序列，通过更多数据的训练，会增强编码器的编码能力。

（3）通过使用同一个编码器，可将输入序列和摘要序列映射到同一向量空间。而解码器后续模块需要在输入序列和摘要序列之间计算注意力权重，两者的向量表示处于同一向量空间，有利于点积注意力函数的计算，使模型能更清楚的挖掘输入序列和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，如公式（1）所示，与编码器中的多头注意力层不同的是，其中$Q = s_{<i}$，而$K = V = \tilde{h}$。前向反馈层用于增加模型的非线性拟合能力，如公式（2）所示。经过这两个层之后，得到输出向量m_i，最终经过 *softmax* 层来预测下一个词，如公式（6）所示：

$$p(y_i|x, y_{<i}) = softmax(m_i W_o + b_o) \tag{6}$$

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。解码器也可堆叠多层，在本文工作中，使用了两层解码器，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。

3.5 目标函数

训练阶段，模型的目标是在给定输入序列 x 后，最大化得到摘要序列 y 的概率，因此目标函数是最小化负对数似然函数，如公式（7）所示：

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

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

4 实验与结果分析

本节介绍了实验所使用的数据集、文本摘要的量化评估指标 ROUGE[14]、模型的实现细节，并定量和定性对比了我们的模型与基线模型的效果。

4.1 数据集

本文使用的数据集为 Gigaword 数据集和 DUC2004 数据集，两个数据集的详细情况如表 1 所示。对于 Gigaword 数据集，我们使用经过 Rush[4]等预处理后的版本，处理细节如下：（1）所有英文单词都小写化；（2）数字以“#”代替；（3）出现次数少于 5 次的单词用未登录词标识符“<unk>”来代替。Gigaword 数据集较大，已被划分为训练集、验证集、测试集三部分，分别包含约 380 万、18 万、1951 个文本-摘要对，用于模型训练和测试。而 DUC2004 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，故本文使用在 Gigaword 数据集上训练好的模型，在 DUC2004 数据集上测试。

表 1　Gigaword 和 DUC2004 数据集的详细情况

Table 1　Details about Gigaword and DUC2004 dataset

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

4.2 评估指标

遵循已有的工作，我们使用 ROUGE 指标来评估生成摘要的质量。ROUGE 指标主要评估模型生成摘要和标准摘要之间重合度，重合度越高，ROUGE 得分也越高。ROUGE 指标从三个粒度来评判重合度，分别为词、二元短语、最长公共子序列，对应三个指标，即 ROUGE-1，ROUGE-2，ROUGE-L，每个指标都包含精确率、召回率、F1 值；使用 F1 值来评判模型在 Gigaword 数据集上的效果，使用召回率来评判模型在 DUC2004 数据集上的效果。

4.3 实现细节

模型参数方面，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，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]，以利用序列的顺序信息，各个网络层之间采用残差连接[15]，以避免梯度消失；使用 dropout 方法[16]来避免过拟合，比率设置为 0.1。

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，每在训练集上训练两次后，将学习率衰减至当前的一半。在训练 10 个 epoch 后，模型趋于收敛。使用一块 Nvidia GTX 1080ti 显卡进行训练。

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，以鼓励模型生成更长的摘要。

4.4 模型说明

本文提出了两个模型，分别为：

ES-Trans：编码器共享的 Transformer (Encoder Sharing Transformer)，在 Vaswani[1]等的工作上进行改进，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：Rush[4]等于 2015 年提出，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：Chopra[6]等于 2016 年提出，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：Nallapati[7]等于 2016 年提出，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，此外使用了多个特征，如词性、实体信息等。

Entail-s2s：Li[18]等于 2018 年提出，基于循环神经网络，引入了门控网络和文本蕴涵任务来进行多任务训练，以增强编码器提取关键信息的能力。

Seq2seq：本文实现的方法，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。

Transformer：Vaswani[1]等提出，原本用于机器翻译，我们实现了该模型并将其用于文本摘要任务。

表 2　在 Gigaword 数据集上各模型的 ROUGE F1 评分（%）及训练时间和测试时间

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

模型	Rouge-1	Rouge-2	Rouge-L	训练时间 (hours/epoch)	推理速度 (items/seconds)
ABS (Rush et al. 2015)	29.55	11.32	26.42	4.2	6.4
Feats2s (Nallapati et al. 2016)	32.67	15.59	30.64	9.8	2.3
RAS (Lin et al. 2016)	33.78	15.97	31.15	5.1	6.6
Entail-s2s (Li et al. 2018)	35.33	17.27	33.19	4.7	7.1
Seq2seq (our implementation)	33.20	15.64	30.82	3.5	8.3
Transformer (our implementation)	33.08	15.37	30.49	**1.3**	**18.3**
ES-Trans.(ours)	35.98	17.30	33.24	1.4	18.1
Gated ES-Trans.(ours)	**36.15**	**17.51**	**33.47**	1.4	17.2

表 3　在 DUC2004 测试集（仅用于测试）上各模型的 ROUGE 召回率评分（%）

Table 3　The ROUGE recall of each model on DUC2004 dataset

模型	Rouge-1	Rouge-2	Rouge-L
ABS (Rush et al. 2015)	26.55	7.06	22.05
Feats2s (Nallapati et al. 2016)	28.35	9.46	24.59
RAS (Lin et al. 2016)	28.97	8.26	24.06
Entail-s2s (Li et al. 2018)	**29.33**	**10.24**	25.24
Seq2seq (our implementation)	27.88	8.41	24.04
Transformer (our implementation)	27.54	8.29	24.01
ES-Trans. (ours)	29.02	9.81	25.12
Gated ES-Trans. (ours)	29.25	9.97	**25.41**

4.5 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, ROUGE-L 的得分，使用封装了 ROUGE 脚本的 pyrouge 工具[1]来计算得分。此外，我们列出了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），用以对比

[1] https://pypi.org/project/pyrouge

模型的时间效率。

表 2 展示了各个模型在 Gigaword 数据集上的 ROUGE 指标的 F1 评分以及其训练时间。本文提出的模型 Gated ES-Trans 在 ROUGE-1、ROUGE-2、ROUGE-L 三个指标上的得分都明显优于其它模型，相对于已有的最优模型 Entail-s2s，得分提升了 0.82、0.24、0.28；而相对于 RAS 模型，得分提升超过 2.0、1.5、2.0，同时训练时间仅为其的 1/3 甚至更少。相比 Transformer 模型，本文基于其进行改进所提出的编码器共享方法 ES-Trans，大幅度提升了 ROUGE 得分，提升值分别达到 2.90、1.93、2.75；此外，相对于已有的生成式方法，本文方法大大缩短了训练时间，提升了推理速度。而加入门控网络后的 Gated ES-Trans 模型，取得了更高的 ROUGE 得分，说明了门控网络的有效性。表 3 展示了各个模型在 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）相对 RAS(Lin, 2016)和 Transformer(Vaswani, 2017)等，ROUGE-2 和 ROUGE-L 得分有明显的提升；相对 Entail-s2s(Li, 2018)，本文方法的效果与其相当，ROUGE-1 和 ROUGE-2 评分稍低，但 ROUGE-L 评分更高。

4.6 案例分析

本节从 Gigaword 测试集中摘取两个例子，展示了本文最终模型与 Vaswani 等的 Transformer 模型在这两个示例上的摘要结果，用以对比这两个方法在生成摘要方面的正确性和完整性的差距。如表 4 所示，每个示例从第一行到第四行分别表示：原文序列、参考摘要、Transformer 模型生成的摘要，本文最终模型 Gated ES-Trans 生成的摘要。

表 4 从 Gigaword 数据集摘选的两个示例，及模型对应的摘要

Table 4 Two examples from Gigaword, and their corresponding summarization

Input：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 .
Reference：sri lanka closes schools as war escalates
Transformer：sri lanka launches campaign against tamil rebels
Gated ES-Trans：sri lanka closes schools as military campaign escalates
Input：**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 .
Reference：nigerian plane crashes on landing killing at least ##
Transformer：at least ## killed in nigeria
Gated ES-Trans：at least ## killed in nigerian airways plane crash

对于表 4 的第一个示例，其主要内容可总结为“斯里兰卡因战争动荡关闭学校”，本文模型产生摘要基本上完整的概括了这一内容，跟参考摘要相差无几。而 Transformer 模型产生的摘要明显提取错了重点，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。对比之下，本文模型产生的摘要更加正确。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。Transformer 模型产生的摘要，正确的说明了人员伤亡情况，但未说明原因，存在信息遗漏；而本文模型则完整地概括了伤亡情况及原因。证明了本文模型生成摘要的正确性以及完整性。

5 结语

本文提出了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，进而增强了对编码器的训练；同时引入了门控网络，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 DUC2004 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。

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, USA, 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, Canada, MIT Press, 2014: 3104-3112

[3] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translation, In Proceedings of 3rd International Conference for Learning Representations, San Diego, USA, 2015

[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, Portugal, 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, Portugal, 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 California, ACL Press, 2016, 93-98

[7] Nallapati R, Zhou Bowe, Cícero Nogueira et al. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, Berlin, Germany, 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, Canada, 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, Australia, ACL Press, 2018: 163-169

[11] Paulus R, Xiong Caiming, Socher R. A Deep Reinforced Model for Abstractive Summarization. 6th International Conference on Learning Representations, Vancouver, BC, Canada, 2018

[12] Nair V, Hinton G E. Rectified Linear Units Improve Restricted Boltzmann Machines. Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, Omnipress, 2010: 807-814

[13] Devlin J, Chang Mingwei, Lee K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, USA, ACL Press, 2019

[14] Lin C. Rouge: A package for automatic evaluation of summaries. Proceedings of the ACL Workshop: Text Summarization Braches Out. Barcelona, Spain, 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, NV, USA, IEEE, 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. In Proceedings of 3rd International Conference for Learning Representations, San Diego, USA, 2015

[18] Li Haoran, Zhu Junnan, Zhang Jiajun, et al. Ensure the Correctness of the Summary: Incorporate Entailment Knowledge into Abstractive Sentence Summarization. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, German, ACL Press, 2016: 1631-1640

[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, Qatar, ACL Press, 2014: 1724-1734