**(2022-2023 学年第二学期)**

# 重庆理工大学研究生课程论文

**课程论文题目：** 基于序列到序列模型的生成式文本摘要研究综述

| | |
|---|---|
| 课 程 名 称 | 社交网络分析及应用 |
| 课 程 类 别 | □学位课 ☑非学位课 |
| 任 课 教 师 | 刘小洋 |
| 所 在 学 院 | 计算机科学与工程学院 |
| 学 科 专 业 | 计算机技术 |
| 姓　　名 | 周涛 |
| 学　　号 | 52220313427 |
| 提 交 日 期 | 2023/6/22 |

**注意事项：**

1、以上各项由研究生认真填写；

2、研究生课程论文应符合一般学术规范，具有一定学术价值，严禁网上下载或抄袭；凡检查或抽查不合格者，一律取消该门课程成绩和学分，并按有关规定追究相关人员责任；

3、论文得分由批阅教师填写（见封底），并签字确认；批阅教师应根据作业质量客观、公正的在文后签写批阅意见；

4、原则上要求所有课程论文均须用 A4 纸双面打印，加装本封面封底，左侧装订；

# 基于序列到序列模型的生成式文本摘要研究综述

周 涛

(重庆理工大学 计算机科学与工程学院，重庆 400054)

摘 要 近年来，互联网信息呈井喷式爆发，如何从中快速有效的获取信息显得极为重要。自动文本摘要技术的出现有效的缓解了该问题。本文梳理了近年来基于序列到序列模型的生成式文本摘要的相关研究，从模型的编码、解码、训练等方面的研究工作分别进行了综述，并对这些工作进行了比较，在此基础上总结出该领域面临的挑战和未来的研究趋势。

关键词 生成式摘要；序列到序列模型；神经网络

## Abstractive Summarization Based on Sequence to Sequence Models: A Review

Tao Zhou

(School of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054)

**Abstract** In recent years, the Internet information has been a blowout explosion, how to obtain information quickly and effectively becomes extremely important. The emergence of automatic text summary technology effectively alleviates this problem. This paper reviews the recent research on sequence-to-sequence model based generative text abstracts, reviews the research work on model coding, decoding, training, and so on, and compares these work. On this basis, it summarizes some technical routes and development directions in this field.

**Key Words** abstractive summarization; Sequence to Sequence model; neural networks

## 1 引言

自 1958 年 Luhn[1]开启了自动摘要研究以来，该邻域已经形成了丰硕的成果。目前，自动摘要方法大体上可以分为两类[2]：抽取式（extractive summarization）和生成式（abstractive summarization）。抽取式的基本做法是从原文中抽取部分重要的句子形成摘要，研究重点集中在句子的重要性判断、筛选以及排序等。生成式摘要的基本思路是在理解原文的基础之上，凝练其中心思想，以实现语义重构。抽取式是之前自动文本摘要研究的重点，不过从近几年的研究结果来看，更多的研究人员将研究重点放在了生成式文本摘要上。

李金鹏等[3]等对止于 2021 年的自动文本摘要的相关研究进行了综述，在其中将生成式文本摘要分为基于结构以及基于语义两类。前者生成摘要的主要不足是语言质量相对较差，比如，语句中包含较多的语法错误；后者生成的摘要具备简明、内聚、信息丰富以及低冗余等优点，不足之

处在于主要使用浅层自然语言处理技术。近年来，深度学习技术为自动文本摘要提供了新的思路，其中，序列到序列（sequence to sequence，Seq2Seq）模型的研究与应用最为广泛。该模型由 Cho 等[4]和 Sutskever 等[5]提出，基本思想是利用输入序列的全局信息推断出与之相对应的输出序列，由编码器（encoder）和解码器（decoder）构成。Rush 等[6]首次将该模型应用于生成式摘要，和先前的生成式方法相比，该模型是在“理解”文本语义的基础上生成摘要，更加接近人工摘要的生成过程。随后，学界提出了一系列基于 Seq2Seq 的生成式摘要模型，对编码器、解码器以及训练方法等开展了卓有成效的研究工作。基于该模型生成的摘要在语言流畅性、连贯性等方面让学界看到自动摘要实用化的希望[7]。

本文第 2 节阐述基础 Seq2Seq 模型，第 3 节按照模型的结构分别梳理编码、解码以及训练等方面的研究进展，第 4 节与第 5 节分别对本领域面临的挑战进行分析与总结。

## 2 序列到序列模型

序列到序列（Seq2Seq）模型基本结构是编码–解码框架，也叫 Encoder-Decoder 模型。蒙特利尔大学 Cho 等[4]对其进行了详细的描述。其最初用于机器翻译任务。生成式自动文本摘要类似于机器翻译任务，都是序列到序列之间的转换。但机器翻译输入序列的长度与输出序列的长度较为接近。自动文本摘要是输入文本序列，然后生成文本的简短描述。源序列和目标序列的长度可以不同。序列到序列模型通过变长序列对之间的映射，可以实现不同领域之间的转化，使输入和输出的长度可变。Seq2Seq 框架视为基于深度学习的通用研究模型之一。它不仅广泛用于自然语言处理领域，而且还广泛用于语音和图像领域。序列到序列（Seq2Seq）模型表示形式如图 2-1 所示，Encoder 的选择可以是任意的模型或数据，Decoder 同样也可以是任意的模型，对于文本摘要来说，传统的模型需要保持输入输出的一致性，该模型的最大特点是输入与输出的序列长度可以不一致。

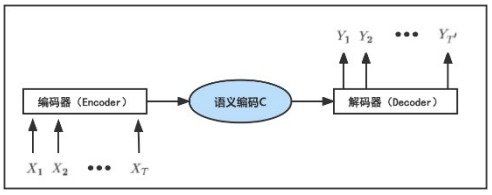


图 2-1 序列到序列（Seq2Seq）模型图

基础的 Seq2Seq 模型包含了编码器、解码器和中间向量 C,编码器通过训练将输入转化为向量 C，解码器在解码阶段，通过对向量 C 的转换，将转换后的单词序列组成后输出，分别为$Y_1$、$Y_2$，直到$Y_{T'}$。

在 Cho 等[4]的工作中,编码器和解码器均采用循环神经网络（recurrent neural network，RNN）。编码器将输入的一个可变长序列$X=(x_1,x_2,\ldots,x_T)$编码为一个固定的语义向量；解码器从该向量中提取语义信息，输出另一个可变长序列$Y=(y_1,y_2,\ldots,y_{T'})$，序列中的每个词项采用词向量表示。模型的具体计算过程如下：编码器基于输入的词向量$x_i$，以及上一词项的隐层$h_{i-1}$，计算当前词项隐层$h_i$[公式(1)]，再通过隐层向量计算语义向量$c$[公式(2)]；解码器在每个时间步$t$，基于语义向量$c$、上一时间步隐层$s_{t-1}$和生成的上一个词项$y_{t-1}$计算当前隐层$s_t$[公式(3)]，再基于语义向量$c$、当前隐层$s_t$，和生成的上一个词项$y_{t-1}$，推导当前词项$y_t$的分布[公式(4)]。

$$h_i=f(x_i,h_{i-1}) \quad (1)$$

$$c=q(\{h_1,h_2,\ldots,h_T\}) \quad (2)$$

$$s_t=f(y_{t-1},s_{i-1},c) \quad (3)$$

$$p(y_t \vee y_{¿t},X)=g(y_{t-1},s_t,c) \quad (4)$$

其中，$f$和$g$为非线性激活函数；$g$通常是 softmax 函数，用于产生词项在词汇表$V$中的概率分布，一般用贪婪算法（greedy search）取最大概率对应的词项作为输出。

模型使用有标注的训练集$D$进行训练，$D$由大量源文本$x$和对应的标准摘要$y$构成。训练的基本目标是优化参数集$\theta$，使输入序列$x$的输出结果最大似然于序列$y$，即最大化 $logp(y \vee x,\theta)$，等同于最小化交叉熵损失，损失函数为

$$L_{MLE}(\theta)=-\sum_{(x,y)\in D} logp(y \vee x,\theta)=¿$$

$$-\sum_{(x,y)\in D}\sum_{t} logp(y_t \vee y_{¿t},x,\theta) \quad (5)$$

该模型的不足之处是，编码器把源文本中的所有信息表示为一个固定的语义向量，解码器在生成每一个词项时均参考该向量，这为神经网络处理长文本带来了困难。Cho 等[4]的实验证实随着文本长度的增加，模型的表现快速下降。对此 Bahdanau 等[8]在模型中引入了注意力（attention）机制,目的是使解码器在生成每一个词项时重点关注源文本中的特定部分，即解码过程不再依赖原先固定的语义向量$c$，而是利用动态的$c_t$[公式(6)~公式(8)]，$c_t$是时间步$t$所有词项隐层的加权。

$$c_t=\sum_{i=1}^{T}\alpha_{ti}h_i \quad (6)$$

$$\alpha_{ti}=softmax(e_{ti})=\frac{\exp(e_{ti})}{\sum_{k=1}^{T}\exp(e_{tk})} \quad (7)$$

$$e_{ti}=score(h_i,s_{i-1}) \quad (8)$$

其中，$e_{ti}$是注意力得分，用来估计位置$i$附近的输入和时间步$t$的输出之间的匹配程度；$a_{ti}$是注意力分布，代表在时间步$t$每个词项的隐层$h_i$被解码器关注的程度。每个输出词项的分布相应地由公式(4)更新为公式(9),

$$p(y_t \vee y_{¿t},X)=g(y_{t-1},s_t,c_t) \quad (9)$$

Bahdanau 等[8]的实验结果表明，带有注意力机制的模型在机器翻译任务上取得了更好的成绩，对于句子长度变化更具鲁棒性。注意力机制的加入使得 Seq2Seq 模型更加完善，之后大量相关研究都建立在该模型的基础上。

# 3 衍化

## 3.1 编码

Rush 等[6]在论文中提出了三种编码器方案：①词袋编码器，将输入序列中的词向量平均后作为语义向量，并不考虑词的顺序；②卷积编码器，使用时滞神经网络（time-delay neural network，TDNN)对词向量交替进行时间卷积( temporal convolution)和最大池化（max pooling）以计算出语义向量；③基于注意力机制的编码器；即 ABS（attention-based summarization）模型。ABS 模型基于词袋编码器,在计算语义向量时，不仅考虑输入序列中的词向量$x$，还考虑解码器已输出的最近$R$个词的向量$y_R^{t-1}$，模型用$y_R^{t-1}$在输入序列和输出序列之间做对齐。

借鉴人类在阅读时会标注出重点内容的做法，Zhou 等[9]提出了选择性编码模型。该模型通过设置一个门控网络对编码器生成的词项隐层进行权重标注，相当于“选择”出相对重要的内容，使解码器可以有针对性的读取源文本。模型的具体实现使用门控循环单元（gated recurrent unit，GRU）计算词项的隐层$h_i=GRU(x_i,h_{i-1})$以及文本表示$h_{sent}$，然后将二者输入基于多层感知机的门控网络以计算出每个词项的权重向量$weight_i$，

$$weight_i=\sigma(h_i,h_{sent}) \quad (10)$$

之后用权重向量更新隐层$h_i$，得到新词项隐层$h_i^{new}$，

$$h_i^{new}=h_i \otimes weight_i \quad (11)$$

其中，$\otimes$代表向量点乘运算( element-wise multiplication )。

Zeng 等[10]提出的“再读（read-again)”模型与 Zhou 等[9]工作类似，不同点是该模型没有直接用权重向量更新当前词项的隐层，而是用另一个 GRU 对源文本进行二次编码，

然后将权重向量用于更新第二次编码生成的词项隐层$h_i^{(2)}$，

$$h_i^{(2)}=\left(1-weight_i\right)\otimes h_{i-1}^{(2)}+¿$$

$$weight_i\otimes GRU^{(2)}\left(x_i,h_{i-1}^{(2)}\right)(12)$$

该模型综合考虑了当前词项的权重、当前隐层$h_i^{(2)}$以及上一个词项的隐层$h_{i-1}^{(2)}$。

Zeng 等[10]还将 GRU 更换为长短时记忆网络（long short-term memory，LSTM）做了对比实验，考虑到 LSTM 使用非线性激活函数来更新隐层，无需单独计算$weight_i$，直接利用第一遍编码获得的词项隐层和文本表示即可更新$h_i^{(2)}$，

$$h_i^{(2)}=LSTM^{(2)}\left(\left[x_i;h_i^{(1)};h_{sent}^{(1)}\right],h_{i-1}^{(2)}\right)(13)$$

实验结果表明，GRU 和 LSTM 的表现不分伯仲。

实验证明，对于 Seq2Seq 模型来说，源文本越长处理难度越大，主要原因在于神经网络的记忆能力有限，即使有注意力机制，也很难联合较远的输入做出判断。因此处理长文本的一个思路是将其拆分成区块（如句子、段落、语篇等），对区块和全文分别进行编码，再用层级注意力( hierarchical attention) 计算语义向量，从而缓解记忆压力，并尝试在语义向量中融入文本的结构特征。

卷积编码即对输入序列进行卷积操作以得出文本表示。Rush 等[6]曾使用基于 TDNN 的卷积编码器计算语义向量，鉴于 TDNN 不擅长处理时间序列，而且模型缺少注意力机制，实验效果并不理想；来自同一团队的 Chopra 等[11]改进了 Rush 等[6]的工作，将输入序列中词项的位置信息嵌入到词向量中，并加入注意力机制，在一定程度上提升了模型的表现。之后的相关工作大多采用更擅长处理时间序列的 RNN 进行建模。2017 年，Gehring 等[12]提出基于 CNN 的卷积序列到序列（convolutional sequence to sequence，ConvS2S）模型，在机器翻译和文本摘要任务中均表现出色，引起学界的关注。

ConvS2S 模型的编码器和解码器均是多层 CNN，编码器对输入序列做多层卷积，解码器在每一层都做注意力计算，即多步注意力（multi-step attention）。模型首先对输入序列中的词项做位置嵌入，将每个词向量$x_i$,和其绝对位置向量$p_i$相加作为模型的输入，即$e=\left(x_1+p_1,\ldots,x_T+p_T\right)$，位置嵌入给原本不擅长处理时间序列的 CNN 带来一些“位置感”。对于第$l$层，用大小为$k$的卷积核$v^{'}\in R^{kd}$对上一层的输出做一维卷积，得到每一层的输出$g^l\in R^{2d}$，其中$d$代表词向量的维度；将$g^l$转换为矩阵$G^l=[G_1G_2]$，其中$G_1,G_2\in R^d$，用门控线性单元（gated linear unit，GLU）对$G^l$做非线性变换；为支持深度卷积网络，在每一层的输出中还增加了残差连接，最后得到第$l+1$层的隐层$h_i^{l+1}$，

$$h_i^{l+1}=G_1^l\otimes\sigma\left(G_2^l\right)+h_i^l(14)$$

多步注意力的具体实现为：首先将解码器第$l$层的隐层$s_t^l$和上一步输出的词项$y_{t-1}$，结合得到$d_t^l$，之后利用$d_t^l$和编码器最后一层的隐层$h_i^L$计算解码器第$l$层的注意力$\alpha_{ti}^l$，

$$\alpha_{ti}^l=\frac{\exp\left(d_t^l\cdot h_i^L\right)}{\sum_{k=1}^{T}\exp\left(d_t^l\cdot h_k^L\right)}(15)$$

最后用$\alpha_{ti}^l$加权$h_i^L$和$e_i$,得到第$l$层的语义向量$c_i^l$，

$$c_i^l=\sum_{i=1}^{T}\alpha_{ti}\left(h_i^L+e_i\right)(16)$$

### 3.2 解码

解码器读取语义向量并输出目标序列，相当于人在理解文本后开始编写摘要。解码过程主要存在以下问题：①当某个词不存在于词汇表中时，便无法生成；②解码器可能会重复关注到源文本的某些部分，导致摘要也产生重复；③解码器变量有限，无法将高级语法或结构信息模型化。围绕上述问题，学界提出多种改进方法。

Zhou 等[9]在 Gigaword 训练集上统计发现，文本摘要中生成的词占 42.5%，其余的均由拷贝所得，且连续两个词以上的拷贝约占 1/3。在之前的拷贝机制中，每次拷贝都有一个决策过程：拷贝还是生成，如果要连续拷贝 3 个词，机器就要做 3 次决策。此，Zhou 等[9]提出序列拷贝网络，其基本思想是如果机器决定要拷贝，则直接拷贝一个子序列，如 3 个词，从而减少决策次数，同时降低了拷贝机制在连续拷贝过程中出错的概率。

Gu 等[13]提出的 CopyNet 在模型结构上有别于先前的拷贝机制，没有使用开关网络和指针网络，在解码时基于生成模式和拷贝模式的混合概率预测单词。模型构造了一个词汇表$X$，只收录存在于输入序列中的词，扩展后的词汇表为$V\cup U$¿$UNK$>¿¿，由于$X$中可能包含不存在于$V$中的词，这部分词将用于拷贝。在输出每个目标单词时分别计算生成模式和拷贝模式的概率，并相加得到混合概率，

$$p\left(y_t\vee s_t,y_{¿t},c_t,H\right)=¿$$

$$p_{gen}\left(y_t\vee s_t,y_{t-1},c_t,H\right)+¿$$

$$p_{copy}\left(y_t\vee s_t,y_{t-1},c_t,H\right)(17)$$

其中，$H$表示由编码器生成的词项隐层$\left[h_1^{'},\cdots,h_T^{'}\right]$构成的矩阵，$h_i^{'}$既包含词项语义信息又包含位置信息。CopyNet 在生成模式下读取语义信息，在拷贝模式下则读取位置信息。由于$H$包含了位置信息，CopyNet 在网络结构上更加简单，不需要开关和指针；但也正是因为$H$的特殊性，限制了 CopyNet 的通用性。

在训练阶段，解码器生成的每个词项都有标准摘要作参考，并将误差反向传播以修正模型参数，因此一般采用贪婪算法取词汇分布的概率最大值作为输出词项。但在测试阶段，没有标准摘要作参考，这时概率最大的词项未必是最好的选择，研究表明，用贪婪算法生成的句子可读性较差[6]。束搜索算法是解决上述问题的一种手段，已被广泛运用于多个摘要模型[6]，具体算法是，给定一个束宽（beam width）B，在解码的每个时间步都保留词汇分布中概率最大的 B 个词项作为候选词项，从第二个时间步开始，会产生 B×B 个候选分支，依然只保留概率最大的前 B 个分支，依次进行下去，直至遇到终止条件。最后得到 B 个候选序列，选择概率最大者作为最终结果。束搜索的问题是缺乏多样性，即 B 个候选序列区别不大，因此 Cibils 等[14]提出多样性束搜索（diverse beam search），将束宽 B 等分为若干个组，在解码时每个组依次进行束搜索，并构造一个差异函数来度量当前组的候选序列和先前组生成的序列之间的差异，通过惩罚差异小的分支以增加组之间生成序列的多样性。

### 3.3 训练

从公式(5)可以看出，基础模型的训练过程属于词级训练，即逐个最大化每个词项的条件概率$p\left(y_t\vee y_{¿t},X\right)$，可能会丢失全局信息。对此,Ayana 等[15]提出最小风险训练（minimum risk training，MRT）策略，属于序列级训练[16]，即通过最小化生成摘要$y^{'}$和标准摘要$y$的距离$\Delta\left(y^{'},y\right)$来估计模型参数，距离$\Delta\left(y^{'},y\right)$利用 ROUGE 值计算而来（如负 ROUGE 值），损失函数为

$$L_{MRT}(\theta)=$$

$$\sum_{(x,y)\in D}\sum_{y'\in Y(x;\theta)} p(y'\vee x;\theta)\Delta(y',y) \quad (18)$$

其中，$Y(x;\theta)$代表训练集中的每个$x$可能生成的摘要的集合。可以看出，最小化$L_{MRT}$可以使模型生成的摘要更加接近标准摘要。

MRT 策略依然属于有监督学习，主要存在两个不足[17]:一是曝光偏差（exposure bias）[16]，由于在训练过程中有真值（ground truth）参考，而测试过程中没有，因此测试时会产生误差累积；二是最大似然于真值并非摘要质量评价的唯一标准。针对以上问题，一些学者使用自我评判（self-critical）[18]：一种强化学习（reinforcement learning，RL)中的策略梯度训练算法来训练模型。模型的训练目标不再似然于真值,而是优化用户定义的度量标准（如 ROUGE）。Paulus 等[17]让模型在每次训练迭代时分别产生两个输出序列：用贪婪算法得到的$\hat{y}$和经随机采样得到的$y^s$。用回报函数$r(\cdot)$返回参数序列和标准摘要$y$相比较得到的 ROUGE 分数。基于强化学习的损失函数为

$$L_{RL}(\theta)=¿$$

$$\sum_{(x,y)\in D}(r(\hat{y})-r(y^s))\,logp(y^s\vee x;\theta) \quad (19)$$

可以看出，最小化$L_{RL}$相当于最大化$y^s$的条件似然，从而增加模型的预期回报。然而，Paulus 等[17]发现强化学习方法虽然可以提高模型的 ROUGE 得分，但生成的摘要在可读性上不如最大似然方法，因此将公式(19)和公式(5)结合，得到混合目标函数，

$$L_{MIXED}=\gamma L_{RL}+(1-\gamma)L_{M\le¿} \quad (20)¿$$

其中，$\gamma$为超参数，用于权衡两个目标的比重。

## 4 挑战及发展趋势

目前，自动文摘技术已应用在某些特定领域。但整体来看，近年大量的工作将研究重点放在了抽取或生成的算法上，数据集与评价指标的研究工作较少。除此之外，关于自动文摘的研究工作缺乏针对性的跨越式进步，还需要突破性的创新工作提升性能才能更广泛地适应各个场景，所以自动文摘任务的质量和性能还面临诸多挑战：

1）数据集。高质量的自动文摘数据集较少，甚至中文长文本数据集缺失[19-20]，限制了中文文本摘要技术的研究。

2）评价指标。自动评价方法过于死板，人工评价方法较主观，缺乏被学术界广泛认可并切实可行的评价方法，这减缓了该任务的发展[21]。

3）语义表达。文档的摘要应有多种表达方式[22-23]，但是目前来说同一语义的不同表达、重复表达同一语义的问题还需要相应的工作来解决。

自动文摘的研究已经有近 60 年的历史，由于该任务的难度导致初期的效果并不理想，随着深度学习的快速发展才使得人们看到自动文摘广泛应用的希望。长期看来，自动文摘的发展有 6 个趋势：

1）数据集。中文、英文和其他语言的高质量自动文摘数据集将有可能推动自动文摘任务的发展[19,24]，若仅依靠人工参与构建数据集将是项耗时耗力的工作，因此如果可以通过计算机自动地构建高质量数据集将是非常有意义的。

2）评价指标。目前有工作提出通过计算文本之间语义相似度、改进的 ROUGE 等对自动文摘进行评价[21],但尚不能有效地扩展，因此更加完善的自动文摘评价指标必然是长期研究的重点问题[25]。

3）方法融合。新技术的探索是永远的话题，对传统算法与深度学习的结合，或抽取式方法与生成式方法进一步融合将是学术界乃至工业界必然的趋势[26-27]。

4）借助外部知识。机器效仿人类生成摘要的过程时需要背景知识的辅助（如纳入背景知识库）[28]，对于深度学习方法来说还可用预训练的模型为自动文摘模型提供强有力的外部知识。

5)弱监督或无监督发展。由于缺乏高质量的自动文摘数据集，一种有效可靠的

方法是通过少量的训练数据或无训练数据使用高效的算法处理自动文摘任务[29]。

6）应用场景。研究人员的重心将会慢慢从普适性的工作转移到特定细分场景上，针对不同的子任务场景提出更加具有针对性的算法，如新闻标题、自动对联、评论摘要、会议摘要、金融快报等[30]。

## 5 总结

Seq2Seq 模型源于机器翻译，文本摘要和机器翻译虽然都属于序列到序列转换问题，但文本摘要聚焦输入序列的关键信息，而且输入和输出之间没有明显的对齐关系[9]，从这一角度来说文本摘要任务更加复杂。尽管 Seq2Seq 模型在机器翻译领域已进入实用阶段[30-31]，但对于文本摘要来说显然还有很长的路要走。随着模型的不断衍化，Seq2Seq 模型生成摘要的方式跟人类思维越来越接近，与此同时生成摘要的质量也越来越好。尽管该模型依然存在很多不足，如难以处理几千词以上的长文本、模型时间复杂度高、样本标注开销大等，但其可以引领生成式文本摘要未来的研究方向。

## 参 考 文 献


[1] Luhn H P. The automatic creation of literature abstracts[J]. IBM Journal of research and development, 1958, 2(2): 159-165.

[2] Gambhir M, Gupta V. Recent automatic text summarization techniques: a survey[J]. Artificial Intelligence Review, 2017, 47(1): 1-66.

[3] 李金鹏,张闯,陈小军,胡玥,廖鹏程.自动文本摘要研究综述[J].计算机研究与发展,2021,58(01):1-21.

[4] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. arXiv preprint arXiv:1406.1078, 2014.

[5] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[J]. Advances in neural information processing systems, 2014, 27.

[6] Rush A M, Chopra S, Weston J. A neural attention model for abstractive sentence summarization[J]. arXiv preprint arXiv:1509.00685, 2015.

[7] Shi T, Keneshloo Y, Ramakrishnan N, et al. Neural abstractive text summarization with sequence-to-sequence models[J]. ACM Transactions on Data Science, 2021, 2(1): 1-37.

[8] Nichol A, Dhariwal P, Ramesh A, et al. Glide: Towards photorealistic image generation and editing with text-guided diffusion models[J]. arXiv preprint arXiv:2112.10741, 2021.

[9] Zhou Q, Yang N, Wei F, et al. Selective encoding for abstractive sentence summarization[J]. arXiv preprint arXiv:1704.07073, 2017.

[10] Zeng W, Luo W, Fidler S, et al. Efficient summarization with read-again and copy mechanism[J]. arXiv preprint arXiv:1611.03382, 2016.

[11] Chopra S, Auli M, Rush A M. Abstractive sentence summarization with attentive recurrent neural networks[C]//Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies. 2016: 93-98.

[12] Gehring J, Auli M, Grangier D, et al. Convolutional sequence to sequence learning[C] //International conference on machine learning. PMLR, 2017: 1243-1252.

[13] Gu J, Lu Z, Li H, et al. Incorporating copying mechanism in sequence-to-sequence learning[J]. arXiv preprint arXiv:1603.06393, 2016.

[14] Cibils A, Musat C, Hossman A, et al. Diverse beam search for increased novelty in abstractive summarization[J]. arXiv preprint arXiv: 1802.01457 , 2018.

[15] Shen S, Zhao Y, Liu Z, et al. Neural headline generation with sentence-wise optimization[J]. arXiv preprint arXiv:1604.01904, 2016.

[16] Ranzato M A, Chopra S, Auli M, et al. Sequence level training with recurrent neural networks[J]. arXiv preprint arXiv:1511.06732, 2015.

[17] Paulus R, Xiong C, Socher R. A deep reinforced model for abstractive summarization[J]. arXiv preprint arXiv:1705.04304, 2017.

[18] Rennie S J, Marcheret E, Mroueh Y, et al. Self-critical sequence training for image captioning[C] Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7008-7024.

[19] Denkowski M, Lavie A. Meteor universal: Language specific translation evaluation for any target language[C]//Proceedings of the ninth workshop on statistical machine translation. 2014: 376-380.

[20] Jiang Y, Bansal M. Closed-book training to improve summarization encoder memory[J]. arXiv preprint arXiv:1809.04585, 2018.

[21] ROUGE L C Y. A package for automatic evaluation of summaries[C]//Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.

[22] Carletta J, Ashby S, Bourban S, et al. The AMI meeting corpus: A pre-announcement[C] International workshop on machine learning for multimodal interaction. Springer, Berlin, Heidelberg, 2005: 28-39.

[23] Fang Y, Zhu H, Muszyńska E, et al. A proposition-based abstractive summariser[C] Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. 2016: 567-578.

[24] Yasunaga M, Kasai J, Zhang R, et al. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks[C]//Proceedings of the AAAI conference on artificial intelligence. 2019, 33(01): 7386-7393.

[25] ROUGE L C Y. A package for automatic evaluation of summaries[C]//Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.

[26] 王毅, 谢娟, 成颖. 结合 LSTM 和 CNN 混合架构的深度神经网络语言模型[J]. 情报学报, 2018, 37(2): 194-205.

[27] Hu B, Chen Q, Zhu F. Lcsts: A large scale chinese short text summarization dataset[J]. arXiv preprint arXiv:1506.05865, 2015.

[28] Wang L, Yao J, Tao Y, et al. A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization[J]. arXiv preprint arXiv:1805.03616, 2018.

[29] Song K, Zhao L, Liu F. Structure-infused copy mechanisms for abstractive summarization[J]. arXiv preprint arXiv:1806.05658, 2018.

[30] 李亚超, 熊德意, 张民. 神经机器翻译综述[J]. 计算机学报, 2018, 41(12): 2734-2755.

[31] 吴飞, 阳春华, 兰旭光, 等. 人工智能的回顾与展望[J]. 中国科学基金, 2018, 32(3): 243-250.

## 批阅教师意见

经综合评价，论文得分为：

**批阅教师签名:**

批阅日期：