计算机科学
COMPUTER SCIENCE

基于多步句子选择-重写模型生成科技文献创新点

许贤哲，陈景强

引用本文

许贤哲，陈景强. 基于多步句子选择-重写模型生成科技文献创新点[J]. 计算机科学，2024，51(10): 344-350.

XU Xianzhe, CHEN Jingqiang. Generation of Contributions of Scientific Paper Based on Multi-step Sentence Selecting-and-Rewriting Model [J]. Computer Science, 2024, 51(10): 344-350.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.230800080

基于多步句子选择-重写模型生成科技文献创新点

许贤哲[1]　陈景强[1,2]
1 南京邮电大学计算机学院　南京 210023
2 江苏省大数据安全与智能处理重点实验室(南京邮电大学)　南京 210023
(cjq@njupt.edu.cn)

摘　要　近年来科技文献数量的显著增加,使得研究人员难以跟上自己所在领域的最新进展。为了保持对前沿研究的追踪,研究者通常依赖于阅读文献中的创新点,该部分简明扼要地概括了关键研究成果。然而,许多作者在文中并未充分地呈现文章的创新内容,这导致读者难以快速掌握研究的核心内容。为了解决这一问题,提出了一个全新的任务,即自动生成科技文献的创新点摘要。该任务的难点之一在于目前缺少相关数据集,于是构建了科技创新点摘要语料库(SCSC)。另一个难点在于目前现有的生成式或抽取式模型在生成创新点方面分别存在冗余度过高和句与句之前缺乏关联性的问题。为了满足生成简洁、高质量创新点的需求,提出了 MSSRsum 模型(一个多步句子选择-重写模型)。最终实验表明,所提模型在 SCSC 和 arXiv 数据集上优于基线模型。

关键词:摘要;科技文献;多步句子选择-重写;生成创新点
中图分类号　TP391

Generation of Contributions of Scientific Paper Based on Multi-step Sentence Selecting-and-Rewriting Model

XU Xianzhe[1] and CHEN Jingqiang[1,2]
1 School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
2 Jiangsu Key Laboratory of Big Data Security & Intelligent Processing(Nanjing University of Posts and Telecommunications), Nanjing 210023, China

Abstract　There has been a significant surge in the number of scientific papers published in recent years, which makes it challenging for researchers to keep up with the latest advancements in their fields. To stay updated, researchers often rely on reading the contributions section of papers, which serves as a concise summary of the key research findings. However, it is not uncommon for authors to inadequately present the innovative content of their articles, making it difficult for readers to quickly grasp the essence of the research. To address this issue, we propose a novel task of contribution summarization to automatically generate contribution summaries of scientific papers. One of the challenges of this task is the lack of relevant datasets. Therefore, we construct a scientific contribution summarization corpus(SCSC). Another issue lies in the fact that currently available abstractive or extractive models tend to suffer from either excessive redundancy or a lack of coherence between sentences. To meet the demand of generating concise and high-quality contribution sentences, we present MSSRsum, a multi-step sentence selecting-and-rewriting model. Experiments show that the proposed model outperforms baselines on SCSC and arXiv datasets.

Keywords　Summarization, Scientific papers, Multi-step sentence selecting-and-rewriting, Generation of contributions

1　引言

近年来,科技文献数量的大幅增加,为研究人员提供了丰富的信息资源。在众多文献中,许多学者在引言部分对其研究进行了创新点的概括,旨在使读者能够迅速掌握论文的核心贡献。然而,早期文献和部分当前的文献缺乏对创新点的明确概括。自动生成创新点方法可以为作者提供初步的草稿,同时也有助于读者快速地了解缺乏概括创新点的文献。

创新点生成与传统的摘要生成[1]、亮点生成[2]有所不同,如图 1 所示,摘要通常采用段落的形式呈现,亮点和创新点则浓缩成几句话。亮点通常由三到四句话组成,通常是直接抽取文章中的重要句子。相比之下,创新点着重强调的是一篇

到稿日期:2023-08-14　返修日期:2024-01-10
基金项目:国家自然科学基金(61806101);江苏省高校自然科学研究项目(21KIB520017)
This work was supported by the National Natural Science Foundation of China(61806101) and Natural Science Foundation of the Jiangsu Higher Education Institutions of China(21KIB520017).
通信作者:陈景强(cjq@njupt.edu.cn)

文章的独特贡献，这些句子通常不是直接从文章中抽取，而是通过要点和关键发现进行总结生成。

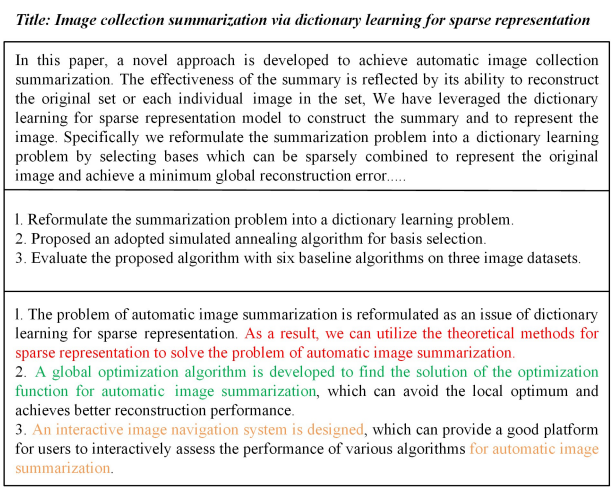

图 1 摘要、重点和贡献之间的区别（电子版为彩图）

Fig. 1 Differences between abstracts, highlights and contributions

为了解决该任务缺乏数据集的问题，本文构建了科技创新点摘要语料库（SCSC）。在此过程中，共汇集了超过 30 万篇科技文献，并通过筛选，整理出大约 17 万份符合实验标准的有效数据。收集过程中发现这些创新点句子往往与科技文献中的摘要和引言中的部分句子有相似之处。因此，我们在构建数据集时保留了这些部分而非全文内容。

对于创新点而言，其真实性和简洁性尤为重要。抽取式方法[3-5]通常保持对源文本的忠实性，但往往会包含冗余信息。而生成式方法可以通过执行句子压缩方法使表达更加简洁。通过观察发现金标准句子与文章本身的某些句子之间存在高度的相似性。因此，本文决定采用选择-重写框架模型，并提出了 MSSRsum，该模型将抽取式和生成式两种方法的优势进行了结合，以达到更好的创新点生成效果。

大多数现有的选择-重写模型都是单步的，这意味着为了生成多个句子，这些单步模型需要独立运行多次，其中上一步生成的结果不会对下一步的选择产生影响。然而，在创新点句子集合中，句与句之间有着很强的相互依赖性，比如存在因果关系、推进关系等。例如，在图 1 中，红色语句和绿色语句之间存在因果关系，橙色语句和前两个句子之间存在递进关系。

与之前的单步策略不同，MSSRsum 采用的是多步选择-重写策略。在任意步中，MSSRsum 通过选择器抽取一个句子，而后选择器不仅将该句子传递给重写器进行重写而且会传递给一个多层感知机网络，以计算停止的概率。为了避免选择相似的句子和增强生成句子之间的连贯性，本文引入了“历史感知”编码器，该编码器会将重写后的句子信息与尚未选择句子的信息进行融合，作为下一步选择器的输入。与使用单步策略的现有方法[6-8]不同，本文提出的方法明确考虑了重写句子对下一个句子选择过程的影响，创新点总结如下：

1）提出了科技论文创新点自动生成的任务，并提出了新的多步选择-重写模型 MSSRsum 来解决该任务。

2）构建了一个新的科技文献领域的数据集（SCSC）。

3）实验表明，根据 ROUGE 评分，模型在 SCSC 和 arXiv 数据集上优于基线模型，这表明了 MSSRsum 的有效性。

2 相关工作

2.1 科技文献摘要

相关工作、引言和摘要的自动生成已成为科技文献生成任务中的热点问题。比如，Chen 等[9]通过检测引文中的常识来生成科技文献摘要，并提出了术语关联发现算法来解决术语不一致的问题。Li 等[10]采用问题和方法信息的优化方法为科技文献生成相关工作部分，该方法使得相关工作部分与原论文主题之间更加贴切。Yu 等[1]采用基于 BART 的方法处理科技文献摘要任务。Chen 等[11]使用比较性引用来识别研究领域内的相关文章，并提出了一种基于图方法的模型，利用引文部分作为指导，生成比较性摘要。Mishra 等[12]使用引文语境化方法提取重要句子，并使用多目标聚类方法对这些句子进行聚类。然而，对创新点部分的研究尚未得到关注。He 等[13]提出了生成科技文献摘要的任务并用该任务来评估它们生成的可控摘要的新框架；Cagliero 等[2]和 Collins 等[14]抽取科技文章中的亮点句子。以上研究给本文提出这项任务带来了启发。

2.2 句子重写模型

句子重写模型是一种基于输入句子自动生成新句子的文本生成模型，这些新句子可以改变原句子的语法结构、表达或语义。Chen 等[8]提出了首个将句子重写模型与强化学习相结合的模型，与以前的模型相比，该模型具有更快的推理速度和更好的训练收敛性。Sanghwan 等[7]改进了 Chen 等[8]的工作，并通过摘要级而不是句子级的奖励，解决了训练目标与评估指标之间不匹配的问题。Xiao 等[6]提出了一种抽取或重写模型，可以灵活地在抽取和重写两种模式之间进行切换，以解决信息丢失的问题。Bao 等[15]通过引入上下文语境的重写方法，解决了提取式摘要存在局限性的问题。该方法将上下文的重写方式形式化为具有组对齐的 seq2seq 任务，并引入组标签作为对齐策略的方案，通过基于标签的寻址来识别提取的句子。与这些单步方法不同，本文提出了一种多步选择-重写方法，该方法将所重写句子的信息考虑到下一步的选择中。

3 构建数据集

我们从互联网上收集了 30 万份 PDF 格式的科技文献并借助 science parse 解析器[1)]将它们转换为 JSON 格式，构建了 SCSC 数据集。在此过程中，成功筛选出 17 万份有效数据，这些数据主要分布在计算机科学和人工智能领域。通过观察发现，摘要和引言部分最有可能准确传达论文的创新点。因此，数据集选择保留每篇论文的这两个部分。此外还使用正则表达式从这些部分中提取创新点。例如，如果在引言的最后一段中找到了类似“创新点总结如下：……”这样的句子，则取

1) https://github.com/allenai/science-parse

2) https://github.com/nltk/nltk

冒号后面的内容，将其选作金标准。在数据清洗方面，共采用3种处理方法。首先使用NLTK[2]对文本进行分词处理；然后利用正则表达式删除停用词和离群值，以确保数据质量；最后剔除不包含摘要、引言或对应创新点句子的文献，以保证数据的准确性和一致性。

表1 SCSC与arXiv，CNN/DM数据集的对比

Table 1 Basic statistics of SCSC compared with arXiv and CNN/DM datasets

	SCSC	arXiv	CNN/DM
数据集大小	175608	215756	300185
句子数/文章	32	206	35
句子数/金标准	3	10	4
字数/文章	973	5206	692
字数/金标准	76	238	49

从表1中可以看出，SCSC中的句子长度明显短于arXiv[16]中的句子长度，与CNN/DM[17]类似。这种差异的产生归因于arXiv数据集包含了完整的科技文献内容，其金标准是摘要部分。相比之下，SCSC只涵盖了论文的摘要、引言和标题部分，并且金标准是相应的创新点句子。

4 MSSRsum模型

鉴于创新点句子与原始文档中的部分句子具有相似性，结合生成式方法的优点，实验中选择采用“选择-重写”的框架，进而提出了MSSRSUM模型。与以往的选择-重写模型不同，MSSRsum采用多步策略最小化创新点句子的冗余度，提高了生成句子间的关联性。

假设文档D由句子集$\{x_1, x_2, \cdots, x_N\}$组成($N$为句子个数)，而$x_i$又由tokens集合$\{x_{i1}, x_{i2}, \cdots, x_{iG}\}$组成($G$为句子中token个数)，模型多步地从$D$中选择一个句子，然后重写这个被选中的句子。金标准$S$由$\{s_1, \cdots, s_t, \cdots, s_T\}$组成，其中$T$表示执行的总时间步数。如图2所示，MSSRsum由编码器、选择器、重写器和基于强化学习的连接器组成。下文将详细介绍各模块。

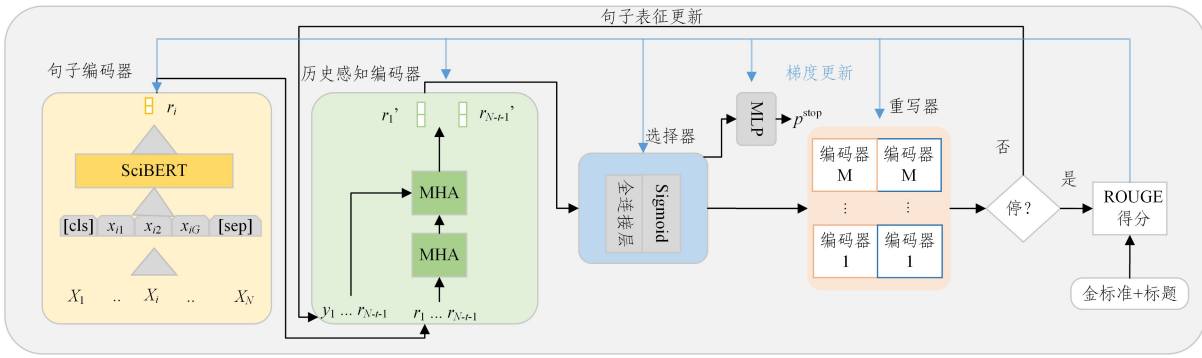

图2 MSSRsum模型的总体架构

Fig. 2 Overall architecture of MSSRsum model

4.1 句子编码器

句子编码器是利用预训练的SciBERT[18]对每个句子进行编码。为了从句子序列中聚合特征，在每个句子之前插入[CLS] token，在其之后插入[SEP] token[19]。第i个[CLS] token对应的向量$\boldsymbol{r}_i$将被用作序列x_i的句子嵌入。

$$\boldsymbol{r}_i = SciBERTEmbedding(x_i) \tag{1}$$

4.2 选择器

选择器被设计成迭代地选择关键句子，然后将其传递给重写器以进行重写和压缩，该过程被称为“多步选择-重写”。考虑到创新点句子之间的相互依赖关系，以及重写后的句子可能会影响下一步选择器的选择，本文还引入了一种反馈机制，即通过历史感知编码器将重写的句子信息融合到剩余未选择句子中。由于未选择句子嵌入包含了之前所选句子的信息，因此可以帮助选择器更好地识别剩余句子，避免选择与之前抽取的相似的句子，其中选择器选择哪个句子是由一种评分策略所决定。此外，模型利用所选句子的信息来学习何时停止选择过程，而非设置固定的句子数量。

4.2.1 历史感知编码器

如图2所示，历史感知编码器由两层多头注意力(Multi-Head Attention，MHA)子层[20]构成。

在第t步中，模型将未选择句子集$\{r_i\}_{i=1}^{N-t+1}$输入第一个MHA子层，以便让这些句子彼此间信息融合。第二个MHA子层使未选择句子集$\{r_i\}_{i=1}^{N-t+1}$能够捕获之前重写句子集$\{y_i\}_{i=1}^{t-1}$的信息。最后，从输出中获得新的嵌入表征，即$\{r_i'\}_{i=1}^{N-t+1}$，以上过程如式(2)所示：

$$\{r_i'\}_{i=1}^{N-t+1} = MHA(MHA(\{r_i\}_{i=1}^{N-t+1}) \bigcup \{y_i\}_{i=1}^{t-1}) \tag{2}$$

4.2.2 评分策略

选择器由包含ReLU激活函数的全连接层和Softmax函数的输出层组成。它的输入是集合$\{r_i'\}_{i=1}^{N-t+1}$，选择器为每个未选择句子计算相应分数，从而得到一个分数向量$\boldsymbol{v}_t$。在$\boldsymbol{v}_t$中，每个元素对应一个句子的分数值，这些值反映了句子在整体上下文中的相关性和重要性。在每一步中，选择器会选择分数值最高的句子作为当前步的选择对象。上述过程如式(3)－式(7)所示：

$$H_t = \{r_t'\}_{i=1}^{N-t+1} \tag{3}$$

$$H_t^l = \mathrm{ReLU}(W_l^{d_2 * d_1} H_t^{l-1} + b_l) \tag{4}$$

$$\boldsymbol{v}_t = \mathrm{Softmax}(W^{1 * d_2} H_t^L + b) \tag{5}$$

$$k = \arg \max_{i=1,\cdots,N-t+1} v_{(t,i)} \tag{6}$$

$$r_t' = H_{(t,k)} \tag{7}$$

其中，$\boldsymbol{W}_l$，$\boldsymbol{b}_l$，$\boldsymbol{W}$，$\boldsymbol{b}$为可训练参数，L为全连接层的层数，$d_2 * d_1$和$1 * d_2$为参数矩阵的大小。

4.2.3 训练选择器

为了使选择器的训练更加有效，实验引入了“代理”标签的概念。对于每个金标准s_t，找到与之最相似的句子r_{j_t}，并

使用式(8)来计算最相似的句子，即选择与s_t有最大 ROUGE-L 分数的句子。

$$j_t=\arg\max_i(ROUGE-L_{\text{recall}}(r_i,s_t)) \tag{8}$$

最后使用这些“代理”标签即 $J=\{r_{j_t}\}_{t=1}^{K}$（K 是创新点句子的数量）来训练选择器以最小化损失函数。

4.3 重写器

对于重写器，实验采用了一个基于 seq2seq 的 SciBERT 模型，该模型已经在 114 万篇科技文献的随机样本上进行了预训练[21]。SciBERT 能够有效捕获上下文的语义信息，特别是在科技领域。本文利用该模型在特定领域的词汇表和对科技文献的上下文理解，增强重写器的文本分析和理解能力。最后，在新数据集(SCSC)上对重写器进行微调，以适应特定的创新点生成任务。另外，为了获取前一步的句子信息，加强前句和后句之间的关系，使用了上一步重写后的句子作为重写器的额外输入，因此重写器的最终输入为句子对，即当前选择句与上步重写句的拼接。

如图 3 所示，重写器由 M 层的编码器和解码器构成。这些编码器和解码器都应用了自注意力机制，以便捕获两个句子之间的相互依赖关系和语义信息。同时，段嵌入用于区分句子对中的两个句子。在图 3 中，y_{t-1} 表示时间步为 $t-1$ 时重写的句子，r_t' 表示在第 t 步中由选择器所选择的句子，y_t 表示在第 t 步中重写的句子。重写器的初始输入 y_0 被初始化为 [PAD] token 所表示的向量。在预训练阶段，金标准 s_{t-1} 可以代替 y_{t-1}，r_{j_t} 可以代替 r_t'。在微调过程中，重写器通过最小化交叉熵损失函数进行微调。

$$\mathcal{L}_{\text{rew}}=-\frac{1}{K}\sum_{t=1}^{K}\sum_{i=1}^{G}(q_{t,i}*\log(p_{j_t,i})) \tag{9}$$

$$p_{j_t,i}\sim\text{Softmax}(H^M) \tag{10}$$

其中，$p_{j_t,i}$ 表示模型预测词的概率，$q_{t,i}$ 表示 s_t 中第 i 个单词的概率分布。H^M 表示重写器中最后一层解码器的隐层状态。

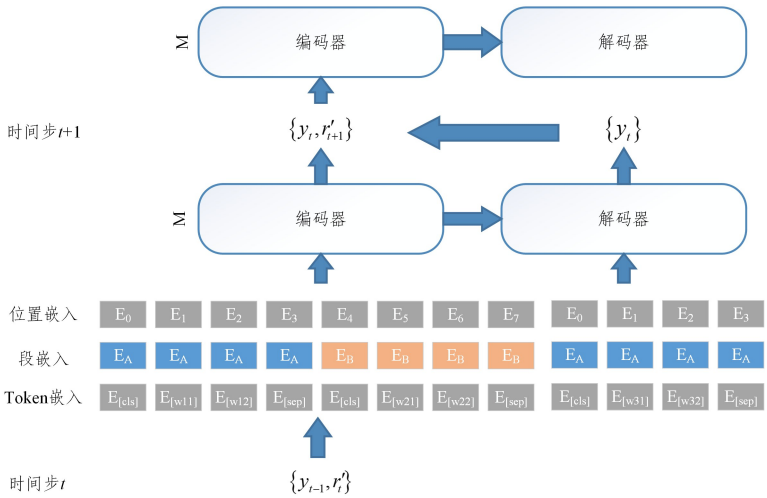

图 3 重写器框架

Fig. 3 Framework of rewriter

4.4 基于强化学习的连接器

基于强化学习(RL)的连接器在选择器和重写器之间充当了桥梁的关键角色，对协调它们的工作起着至关重要的作用。与基于 RL 的传统方法选择固定数量的句子不同，该模型包含了一个停止机制，允许模型根据已选句子的状态动态地决定是否停止。连接器的目标是最小化目标函数 $\mathcal{L}(\theta)=-\mathbb{E}_{\pi_\theta}(R)$，梯度策略可以定义如下：

$$\nabla\mathcal{L}(\theta)=-\mathbb{E}_{\pi_\theta}[R_t\sum_{t=0}^{T}\nabla\log\pi_\theta(A_t|S_t,\theta_t)] \tag{11}$$

其中，$\pi_\theta(A_t|S_t,\theta_t)$ 表示在状态为 S_t 时，使用策略网络 π_θ 执行动作 A_t 的概率，其中 S_t 表示 t 时刻的句子集状态。选择器为剩下的每个句子生成一个输出分数，并计算停止概率。在此过程中，动作 A_t 可以停止选择，也可以进行抽取。以上过程可以表示为式(12)－式(14)：

$$\pi_\theta(A_t|S_t,\theta_t)=p(stop|S_t,\theta_t)*p(A_t|stop,S_t,\theta_t) \tag{12}$$

$$p(stop|S_t,\theta_t)=p_t^{\text{stop}}=MLP(r'_t) \tag{13}$$

$$p(A_t|stop,S_t,\theta_t)=\begin{cases}v(t,i), & p_{t-1}^{\text{stop}}\leqslant p^{\text{gthre}}\\ \dfrac{1}{N-t+1}, & p_{t-1}^{\text{stop}}>p^{\text{thre}}\end{cases} \tag{14}$$

停止机制的概率用式(13)计算，其中 p_t^{stop} 是由选择器后的多层感知机计算得到的标量值，表示停止概率。重写后，将 p_t^{stop} 与超参数 p^{thre} 进行比较，以确定模型是继续下一步抽取还是停止。式(14)表示，如果 $p_{t-1}^{\text{stop}}\leqslant p^{\text{thre}}$，那么状态 S_{t-1} 将首先更新到 S_t，使 $\{r_i\}_{i=1}^{N-t+1}$ 融合 $t-1$ 步中重写的句子的信息。然后，输出一个新的概率 p^{stop} 并把选择的句子向量输出到重写器进行重写，如果 $p_{t-1}^{\text{stop}}>p^{\text{thre}}$，$p(A_t|stop,S_t,\theta_t)$ 将被设置为 $1/(N-t+1)$，这意味着动作 A_t 抽取剩余句子的概率是均匀分布的。在这种情况下，梯度下降将停止，也就表示停止抽取。如果 $p_{t-1}^{\text{stop}}\leqslant p^{\text{thre}}$，那么 $p(A_t|stop,S_t,\theta_t)$ 将被设置为 $v(t,i)$。

通常在基于 RL 的生成摘要应用中[3]，除了最后一个时间步 R_T 外，其余时间步的奖励 R_t 都设置为 0，因此 $R\equiv R_T$。在结束时，通过比较重写句子集和金标准集的 ROUGE 分数[22]计算奖励值 R。

通过观察发现，文献的标题通常也反映文章的创新，特别是标题中的名词短语，因此在奖励计算中考虑了标题信息，如式(15)所示：

$$R=\frac{1}{3}\left[R-1+\frac{1}{2}(R-2+R-2_{\text{title}})+R-L\right] \tag{15}$$

5 实验

5.1 实验设置

实验采用 SCSC 和 arXiv 这两个科技文献数据集，数据集分割信息如表 2 所列。

表 2 数据集分割

Table 2 Dataset splitting

	SCSC	arXiv
训练集	140000	202880
验证集	24000	6436
测试集	11608	6440

实验使用 Adam 最优化算法[23]进行了 100 次迭代训练，其中 Adam 的 β_1 和 β_2 参数分别设置为 0.9 和 0.999。学习率 α 被设置为固定值 1×10^{-4}，权重衰减值为 1×10^{-5}，丢弃正则化设置为 0.1，训练执行的批量大小为 24，当模型在验证集上的性能开始下降时，则停止训练。模型在 RTX 2080Ti GPU 进行训练。实验使用的预训练 SciBERT 模型提供了 768 维度的词嵌入，由于[CLS] token 表示句子，因此句子向量的维度也是 768。实验将重写器中编码器和解码器的层数设置为 6($M=6$)并对其进行了微调，微调时批量大小同样设置为 24，另外还将多层感知机的层数设置为 2($L=2$)，其隐藏层大小分别设置为 64 和 1。对于验证数据集，每个文档的最大句子数 N_{max} 在 SCSC 数据集上设置为 4，arXiv 上设置为 5。阈值 p^{thre} 在 arXiv 和 SCSC 数据集上分别设置为 0.5 和 0.6，详见 5.5 节。

本文使用根据金标准计算的不同 ROUGE 评分(ROUGE-N：基于 n-gram 重叠的评估指标)[22]来评估模型的性能，包括 ROUGE-1，ROUGE-2 和 ROUGE-L。

5.2 基准方法

本文选择 10 种基准方法与 MSSRsum 进行比较，具体介绍如下：

Lead-3 选择文本中的前 3 个句子作为摘要。

LexRank[24]是用基于图的方法选择关键句子。

rnn-ext+RL[8]将 RL 应用于 pointer network[25]来提取重要句子。

MemSum[26]是一个基于马尔可夫链的提取器。

Pointer-generator[27]结合了指针网络和生成网络的优点来处理复杂的文本摘要任务。

Ctrlsum[13]允许用户通过关键字或提示与系统交互来控制生成摘要的细节。

对 SciBERT[18]进行微调，用于文本摘要。

Sentence rewriting[8]采用指针网络进行选择，生成器网络进行改写。

Two-Stage BERT[28]是一种基于预训练的编码器-解码器框架，它基于输入序列以两阶段的方式生成输出序列。

Copy-or-rewrite[6]可以根据冗余程度在抽取或重写模式之间灵活切换。

5.3 对比结果

表 3 列出了模型在 SCSC 和 arXiv 数据集上的结果。实验结果显示抽取式方法和抽取-生成式(基于 RL)方法优于生成式方法，原因在于科技数据集中金标准通常与科技文献主体中的句子具有类似的特点。在抽取式中，使用 RL 的 rnn-ext+RL[8]和 MemSum[27]优于 Lead-3 和 LexRank[25]，这说明了强化学习在句子选择中的有效性。在此任务中，采用抽取-生成框架的混合方法表现出优越的性能，其中与 Copy-or-rewrite 和 Two-Stage BERT 这两个最新的混合模型相比，MSSRsum 在该任务中的表现更好，这可以归因于：1)多步策略的优势，它考虑了前一步结果对后续步骤的选择或生成的影响；2)MSSRsum 在科技领域的效率更高，其能够更好地捕捉科技文本的特征信息。

表 3 MSSRsum 和基线模型在 SCSC 和 arXiv 数据集上的结果

Table 3 Results of MSSRsum and baselines on SCSC and arXiv datasets

	模型	SCSC			arXiv		
		ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
	ORACLE	56.9	29.8	49.3	60.1	33.8	50.4
抽取式	Lead-3	35.3	12.2	30.4	34.0	10.3	29.8
	LexRank	37.1	15.5	33.2	37.6	14.7	31.4
	Rnn-ext+RL	42.3	18.5	38.3	42.3	18.5	38.2
	MemSum	43.3	19.2	38.5	48.4	20.3	41.4
生成式	Pointer-generator	37.6	17.3	35.7	45.7	17.2	35.6
	Ctrlsum	38.7	16.9	36.5	46.8	18.1	37.7
	SciBERT	38.5	17.9	37.2	46.5	17.5	38.1
抽取-生成式(基于 RL)	Sentence rewriting	41.4	18.7	38.3	46.4	20.2	39.3
	Two-Stage BERT	42.7	19.3	38.5	47.4	20.4	39.0
	Copy-or-rewrite	43.2	19.5	**39.1**	48.2	20.6	**41.5**
	MSSRsum	43.8	19.9	38.8	**48.8**	**21.0**	41.2

注：加粗表示最佳分数，下划线表示次优分数。

5.4 消融实验

为了评估模型中重写器、SciBERT 句子嵌入和多步策略的有效性，我们在 SCSC 数据集进行了消融实验以衡量每个组件对模型性能的影响。

实验结果如表 4 所列，从中可以明显观察到每个模块对模型的性能都发挥着至关重要的作用，而将它们组合在一起才能获得最佳性能。表 4 第一行为 MSSRsum 模型，在消融实验中取得了最佳性能；第二行对比了去除了重写器的 MSSRsum 模型的实验结果；第三行呈现了将句子编码器中的预训练 SciBERT 替换为 BERT 模型的实验结果；最后一行则展示了采用单步策略替代多步策略的 MSSRsum 模型的实验结果。

表 4 模型在 SCSC 数据集上的消融实验

Table 4 Ablation study on SCSC

模型	ROUGE-1	ROUGE-2	ROUGE-L
MSSRsum	**43.8**	**20.2**	**38.8**
-重写器	43.2	19.5	38.3
-SciBERT 句子嵌入 +BERT 句子嵌入	42.2	18.4	36.7
一多步策略 +单步策略	42.7	19.8	38.2

5.5 选择最佳的停止阈值

停止阈值p^{thre}是一个重要的超参数，它表示模型的停止概率，如第 4 章所述。实验选择最佳停止阈值的方法如下：对于停止阈值$p^{\text{thre}} \in \{0.1, 0.2, \cdots, 1.0\}$，选择相应验证集上 ROUGE-L 最大的$p^{\text{thre}}$作为最佳停止阈值。

在 arXiv 和 SCSC 的验证集上，ROUGE-L 分数作为停止阈值的函数如图 4 所示。函数显示出 0.1～1.0 之间的局部最大值，这意味着当p^{thre}太低时，摘要往往太短，而当阈值太高时，摘要会过于冗长。因此我们选择当 ROUGE-L 刚好达到最大时的p^{thre}作为对应数据集的阈值，在 arXiv 数据集上将阈值设置为 0.5，在 SCSC 上设置为 0.6。

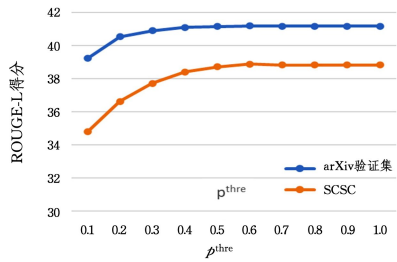

图 4 在 arXiv 和 SCSC 数据集上不同阈值p^{thre}的 ROUGE-L 分数

Fig. 4 ROUGE-L scoresat different thresholds p^{thre} on arXiv and SCSC datasets

5.6 人工评测

最后使用人工评价来验证以上实验的实际有效性。将 MSSRsum 与以往效果最好的模型 Copy-or-rewrite[6] 在冗余度、关联性、真实性和可读性这 4 个因素上进行比较。其中每个因素都以 1－5 的等级进行评分，较高的分数表示对应因素效果更佳(冗余度相反)。为了进行评估，从 SCSC 测试集中随机选择 30 个样本，并由 3 名工作人员独立地对生成的创新点摘要进行评估。每个工作人员根据他们对结果的判断，对上述因素进行评分。然后取分数的平均值，最终结果如表 5 所列。

表 5 人工测评

Table 5 Manual evaluation

模型	冗余度	关联性	真实性	可读性
Copy-or-rewrite	2.9	3.1	3.7	3.9
MSSRsum	**1.8**	**4.2**	**3.9**	**4.1**

结束语 本文提出了自动生成科技文创新点句子的新任务。构建了 SCSC 数据集，其中包含约 17 万篇科技文献及其相应的创新点，此外还引入了 MSSRsum，这是一种利用 RL 连接器集成选择器-重写器的混合方法，能够通过动态多步策略有效地重写选定的重要句子。其中多步策略是由一个灵活的停止机制决定的，而不是一个预先确定的固定值。此外，在此策略中还使用了一个具有历史感知的编码器，以达到加强一致性和减少冗余度的目的。在 SCSC 和 arXiv 数据集上的实验结果表明本文提出的模型优于基线方法，这些结果证明了该模型和数据集在科技文献摘要生成领域的有效性。由于本文提出的模型是在英文数据集上进行预训练和训练，因此它并不适用于中文文献。完善中文文献的数据集以及提升模型在中文文献上的推理性能是我们下一步研究的重点。

参 考 文 献

[1] YU T Z, SU D, DAI W L, et al. Dimsum @LaySumm 20[C]// Proceedings of the First Workshop on Scholarly Document Processing, Online: Association for Computational Linguistics. 2020: 303-309.

[2] CAGLIERO L, LA QUATRA M. Extracting highlights of scientific articles: A supervised summarization approach [J]. Expert Systems with Applications, 2020, 160: 113659.

[3] NARAYAN S, COHEN S B, LAPATA M. Ranking sentences for extractive summarization with reinforcement learning[C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: Association for Computational Linguistics, 2018: 1747-1759.

[4] ZHANG S Y, DAVID W, MOHIT B. Extractive is not Faithful: An Investigation of Broad Unfaithfulness Problems in Extractive Summarization[C]// Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. Toronto, Canada: Association for Computational Linguistics, 2023: 2153-2174.

[5] AKANKSHA J, EDUARDO F, ENRIQUE A, et al. DeepSumm: Exploiting topic models and sequence to sequence networks for extractive text summarization [J]. Expert Systems with Applications, 2023, 211: 118442

[6] XIAO L, WANG L, HE H, et al. Copy or rewrite: Hybrid summarization with hierarchical reinforcement learning [C]// Proceedings of the AAAI Conference on Artificial Intelligence. New York: AAAI, 2020: 9306-9313.

[7] SANGHWAN B, TAEUK K, JIHOON K, et al. Summary Level Training of Sentence Rewriting for Abstractive Summarization [C]// Proceedings of the 2nd Workshop on New Frontiers in Summarization. Hong Kong, China: Association for Computational Linguistics, 2019: 10-20.

[8] CHEN Y C, BANSAL M. Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting[C]// Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne, Australia: Association for Computational Linguistics, 2018: 675-686.

[9] CHEN J, ZHUGE H. Summarization of scientific documents by detecting common facts in citations[J]. Future Generation Computer Systems, 2014, 32: 246-252.

[10] LI P, LU W, CHENG Q. Generating a related work section for scientific papers: an optimized approach with adopting problem and method information [J]. Scientometrics, 2022, 127(8): 4397-4417.

[11] CHEN J Q, CAI C X, JIANG X R, et al. Comparative graph-based summarization of scientific papers guided by comparative citations[C]// Proceedings of the 29th International Conference on Computational Linguistics. Gyeongju, Republic of Korea: International Committee on Computational Linguistics, 2022: 5978-5988.

[12] MISHRA S K, SAINI N, SAHA S, et al. Scientific document summarization in multi-objective clustering framework [J]. Applied Intelligence, 2022, 52(2): 1520-1543.

[13] HE J X, KRYSCINSKI W, MCCANN B, et al. CTRLsum: Towards Generic Controllable Text Summarization[C]// Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, 2022: 5879-5915.

[14] ED C, ISABELLE A, SEBASTIAN R. A Supervised Approach to Extractive Summarisation of Scientific Papers[C]// Proceedings of the 21st Conference on Computational Natural Language Learning. Vancouver, Canada: Association for Computational Linguistics, 2017: 195-205.

[15] BAO G, ZHANG Y. Contextualized rewriting for text summarization [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2023, 31: 1624-1635.

[16] ARMAN C, FRANCK D, DOO S K, et al. A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents[C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics. New Orleans, Louisiana: Association for Computational Linguistics, 2018: 615-621.

[17] NALLAPATI R, ZHOU B W, DOS SANTOS C, et al. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond [C]// Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning. Berlin, Germany: Association for Computational Linguistics, 2016: 280-290.

[18] IZ B, KYLE L, ARMAN C. SciBERT: A Pretrained Language Model for Scientific Text[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong, China: Association for Computational Linguistics, 2019: 3615-3620.

[19] NILS R, IRYNA G. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks[C]// Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. Hong Kong, China: Association for Computational Linguistics, 2019: 3982-3992.

[20] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems. California: Neural Information Processing Systems, 2017: 6000-6010.

[21] WALEED A, DIRK G, CHANDRA B, et al. Construction of the Literature Graph in Semantic Scholar[C]// Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans-Louisiana: Association for Computational Linguistics, 2018: 84-91.

[22] LIN C Y. ROUGE: A package for automatic evaluation of summaries[C]// Text Summarization Branches Out. Barcelona, Spain: Association for Computational Linguistics, 2004: 74-81.

[23] KINGMA D P, BA J. Adam: A method for stochastic optimization[J]. CoRR, 2014, 1412: 6980.

[24] ERKAN G, RADEV D R. Lexrank: Graph-based lexical centrality as salience in text summarization [J]. Journal of Artificial Intelligence Research, 2004, 22: 57-479.

[25] VINYALS O, FORTUNATO M, JAITLY N. Pointer networks [C]// Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal: Neural Information Processing Systems, 2015: 2692-2700.

[26] GU N, ASH E, HAHNLOSER R H. MemSum: Extractive Summarization of Long Documents Using Multi-Step Episodic Markov Decision Processes[C]// Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. Dublin, Ireland: Association for Computational Linguistics, 2022: 6507-6522.

[27] 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. Vancouver, Canada: Association for Computational Linguistics. 2017: 1073-1083.

[28] ZHANG H, CAI J, XU J, et al. Pretraining-based natural language generation for text summarization[C]// Proceedings of the 23rd Conference on Computational Natural Language Learning(CoNLL). Hong Kong: Association for Computational Linguistics, 2019: 789-797.

XU Xianzhe, born in 1999, postgraduate. His main research interests include text summarization and natural language processing.

CHEN Jingqiang, born in 1983, Ph. D, associate professor. His main research interests include text summarization and natural language processing.

(责任编辑:何杨)