计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(11):246−253 [doi: 10.15888/j.cnki.csa.008745]

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

基于关键词与指针生成网络的摘要生成算法①

邓珍荣, 汤园钰, 杨　睿, 张永林

(桂林电子科技大学 计算机与信息安全学院, 桂林 541004)
通信作者: 汤园钰, E-mail: 1778672797@qq.com

摘　要: 为解决传统生成式模型在生成摘要的过程中会忽略关键词信息为摘要提供的重要线索, 导致关键词信息的丢失, 生成的摘要不能很好地契合原文信息, 文章提出了一种以指针生成网络为骨架融合 BERT 预训练模型和关键词信息的摘要生成方法. 首先, 结合 TextRank 算法与基于注意力机制的序列模型进行关键词的提取, 使得生成的关键词能够包含更多的原文信息. 其次, 将关键词注意力加入到指针生成网络的注意力机制里, 引导摘要的生成. 此外, 我们使用双指针拷贝机制来替代指针生成网络的拷贝机制, 提高拷贝机制的覆盖率. 在 LCSTS 数据集上的结果表明, 所设计的模型能够包含更多的关键信息, 提高了摘要生成的准确性和可读性.

关键词: 文本摘要; 关键词; 指针生成网络; 注意力机制; 双指针; 深度学习

引用格式: 邓珍荣,汤园钰,杨睿,张永林.基于关键词与指针生成网络的摘要生成算法.计算机系统应用,2022,31(11):246–253. http://www.c-s-a.org.cn/1003-3254/8745.html

Summarization Algorithm Based on Key Words and Pointer Generation Network

DENG Zhen-Rong, TANG Yuan-Yu, YANG Rui, ZHANG Yong-Lin

(School of Computer and Information Security, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: The traditional generative model ignores the important clues provided by key words in the process of abstract generation, which leads to the loss of key word information, and the generated abstract cannot agree with the original text well. In this study, an abstract generation method is proposed, which takes the pointer-generator network as the framework and integrates BERT pretraining model and key word information. Firstly, the TextRank algorithm and the sequence model based on the attention mechanism are used to extract key words from the original text, and thus the generated key words can contain more information about the original text. Secondly, the key word attention is added to the attention mechanism of the pointer-generator network to guide the generation of an abstract. In addition, we use the double-pointer copy mechanism to replace the copy mechanism of the pointer-generator network and thus improve the coverage of the copy mechanism. The results on LCSTS data sets reveal that the designed model can contain more key information and improve the accuracy and readability of generated abstracts.

Key words: text summarization; key words; pointer generation network; attention mechanism; double pointer; deep learning

1　引言

自动文本摘要目的是生成一个包含原文主要信息且简洁的文本. 自动摘要主要分为抽取式和生成式. 提取式是选择原文一些重要的句子组成摘要, 生成式是计算机在理解原文主旨后生成含有创造性句子的摘要.

近些年随着计算机硬件的发展和大量数据集的出现, 为构建优秀的摘要生成模型提供了可能性. Google Brain 团队[1] 在 2014 年提出的序列对序列模型被成功应用到自然语言处理任务中, 解决了自然语言处理中翻译的问题. 随后 Rush 等人[2] 首次在生成式摘要方法

① 基金项目: 广西科技计划 (AB20238013)
收稿时间: 2022-01-28; 修改时间: 2022-02-24; 采用时间: 2022-03-03; csa 在线出版时间: 2022-07-07

中使用序列到序列模型将原文映射成摘要, 编码器将输入原文编码, 解码器使用编码器得到的上下文向量计算出最终的摘要, 结果表明, 该模型在 DUC-2004[3] 和 Gigaword[4] 数据集上取得了不错的成绩. 为了解决传递依赖, Chopra 等人[5] 将序列到序列模型的解码器用循环神经网络替换, 提高原文词与词之间的依赖关系, 在 Gigaword 数据集上提高了模型的准确性. 以上方法表明了序列到序列模型在生成式文本摘要中的效果, 但生成的摘要存在许多未登录词和句子之间不连贯的问题. 为解决摘要中的未登录单词和可读性差的问题, Gu 等人[6] 提出了 CopyNet 模型, 当生成单词为未登录词时可以通过拷贝机制来进行子序列的拷贝, 该模型在中文数据集 LCSTS[7] 上有效地减少未登录词的数量, 提高了模型的可读性. See 等人[8] 改进了拷贝机制, 提出了使用指针来选择复制原文单词或者生成词汇表单词的指针-生成模型, 该模型减少了生成摘要的重复问题.

2 相关工作

指针生成网络能够捕获文本生成的规律, 但生成过程存在不可控性, 导致生成的摘要不够专注于原文的主要信息, 最终摘要效果不理想. 在没有外界的引导的情况下, 指针很难识别原文的关键字. 为解决这些问题, 本文提出一种融入关键词信息的指针生成网络模型. 首先, 使用 BERT 预训练模型获取文章的多维语义特征的向量和使用提取模型进行关键词的提取, 我们将关键词提取模型看成原文到关键词的映射, 结合 TextRank 算法[9] 进行关键词的提取. 然后, 我们使用提取的关键词信息与原文进行注意力的计算, 该注意力加入到指针生成网络的注意力机制里, 使得生成的摘要更加关注原文的信息. 此外, 使用双指针在关键词信息和原文中进行单词的复制, 提高了模型的准确性和可读性.

基于序列到序列的摘要生成方法会导致生成的摘要丢失关键信息. 为了突出关键词信息对摘要生成的重要性, 许多研究利用关键词信息来提高模型生成摘要的质量. 巴志超等人[10] 通过论文属性加权进行关键词的选择, 用词模型的语义相关性度量方法来衡量关键词的重要性, 结果表明了关键词对生成摘要的有效性. Wan 等人[11] 在假设摘要和关键词可以相互增强的前提下, 从单个文档中同时提取摘要和关键词, 改善了摘要的质量. 近几年, Li 等人[12] 提出了将抽取法和抽象法相结合的引导生成模型, 利用关键词计算注意分布来引导摘要的生成. Li 等人[13] 采用关键字引导的选择性编码策略, 通过研究输入句子和关键字之间的相互作用来过滤源信息, 在英文数据集上取得了很好的效果. 上述方法表明关键词在英文数据集有利于文本摘要的生成. 在本文中, 为了突出关键词的重要性, 我们提出了一种新的关键词信息与指针生成网络的融合策略, 让关键词信息引导指针生成网络进行摘要的生成.

3 KAP 模型

在本节中, 我们将描述关键词与指针生成网络 (key words and pointer-generator, KAP). 图 1 是 KAP 的网络模型图, 它由关键词信息、编码器、解码器组成. 在 KAP 中我们把每一步的关键词信息融入到注意力机制中, 引导摘要生成, 同时使用双指针进行单词的复制, 提高单词复制的准确性.

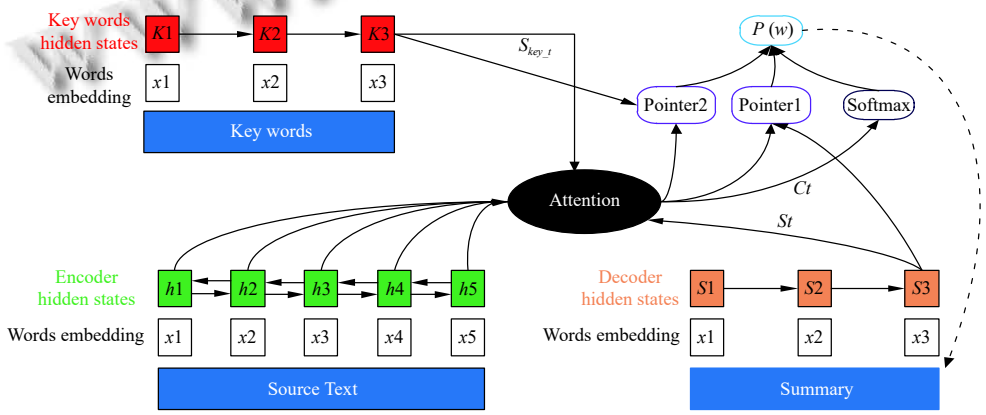

图 1 KAP 网络模型图

3.1 指针生成网络

我们的基线模型是 See 等人[8]提出的指针生成网络模型, 文章 $X=\{x_1, x_2, \cdots, x_n\}$输入到编码器中, 编码器将原文映射成隐藏状态 $h_i=\{h_1, h_2, \cdots, h_n\}$, 解码器的输出状态为 S_t 和 h_i 进行注意力的计算, 注意力机制计算公式为:

$$e_i^t = V^{\mathrm{T}} \tanh(W_h h_i + W_s S_t + b_{\mathrm{attn}}) \quad (1)$$

$$a^t = Softmax(e^t) \quad (2)$$

其中, V^{T}、W_h、W_s、b_{attn} 为可学习参数. 之后注意力分布用于生成语义向量:

$$h_i^* = \sum_i a_i^t h_i \quad (3)$$

将式 (3) 得到的语义向量与解码器状态 S_t 连接, 通过线性层后生成词汇分布 P_{vocab}:

$$P_{\mathrm{vocab}} = Softmax(V'(V[S_t, h_t^*] + b) + b') \quad (4)$$

其中, V'、V、b、b'是模型学习参数. 式 (4) 得到 P_{vocab} 为词汇表单词的概率, 我们通过式 (6) 计算单词 w 的最终分布:

$$P(w) = P_{\mathrm{vocab}}(w) \quad (5)$$

指针生成网络通过式 (6) 得到的指针来选择从原文中复制单词还是从词汇表生成单词. 根据式 (3) 计算的语义向量 h_t^*计算指针, 计算公式为:

$$P_{\mathrm{gen}} = \sigma(w_{h^*}^{\mathrm{T}} h_t^* + w_s^{\mathrm{T}} s_t + w_x^{\mathrm{T}} x_t + b_{ptr}) \quad (6)$$

其中, w_{h^*}、w_s、w_x、b_{ptr} 是模型学习参数. 最终通过式 (7) 计算出预测单词 w 的分布:

$$P(w) = P_{\mathrm{gen}} P_{\mathrm{vocab}}(w) + (1 - p_{\mathrm{gen}}) \sum_{i:w_i=w} a_i^t \quad (7)$$

指针生成网络的具体流程如图 2 所示. 原文与摘要的隐藏层计算注意力机制, 之后使用注意力机制计算出语义向量, 语义向量和摘要的隐藏层计算出 P_{gen}, 使用 P_{gen} 来计算出最终词汇分布.

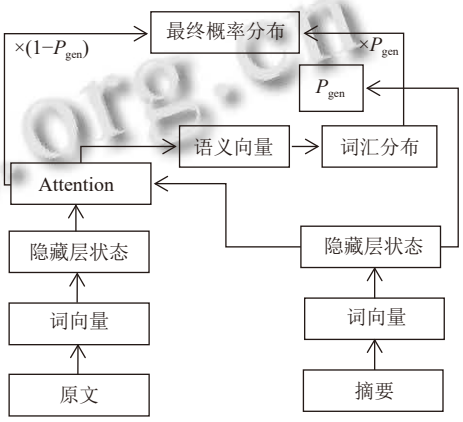

图 2 指针生成网络流程图

3.2 基于序列到序列的关键词提取模型

对于关键词提取任务, 传统的词频进行统计的方法不能很好地反应关键词的信息. 本文使用基于注意力机制的序列到序列模型, 将原文作为模型的输入, 关键词作为模型的输出. 关键词抽取模型如图 3 所示.

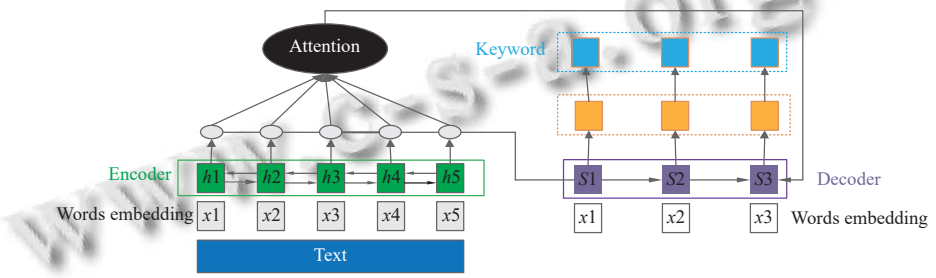

图 3 关键词提取模型

模型先把每个单词转化为词向量, 编码器通过 LSTM (长短时记忆网络) 接受每个单词的词向量和上个时刻的隐藏层状态, 计算出语义向量, 将得到的语义向量和解码器的隐藏层状态融合, 计算出关键词输出的概率. 使用基于注意力机制的序列到序列模型提取的关键词能够更好地反映原文的主要信息, 例如一些人名、地名等情况, 当这些词作为关键词时, 该模型也可以很好地提取到. 关键词提取流程如图 4 所示.

为提高关键词的质量, 我们使用 TextRank 算法[9]二次提取原文的关键词. TextRank 是使用图模型的排序算法, 该算法先将原文中的每个单词形成图模型, 然后词与词之间进行投票选出重要的关键词. 我们结合 TextRank 算法与基于注意力机制的序列模型提取的关键词, 融合的基本思想: 优先选择两者提取的相同关键

词, 其次选择基于序列到序列提取的关键词, 最后选择 TextRank 提取的关键词.

3.3 KAP 网络模型

KAP 网络模型结合了关键词信息和指针生成网络, KAP 网络流程图如图 5 所示. 基于编码器-解码器模型只是将原文作为输入, 摘要作为输出, 在摘要生成的过程中很难把握关键信息, 导致生成的摘要不够契合原文的信息. 我们从关键词出发, 在生成摘要的过程中凸显关键词信息的重要性, 达到关键词引导网络的目的, 以改善生成摘要的质量.

KAP 网络模型首先使用 BERT[14] 来进行预训练, BERT 的编码器为双向 Transformer 结构, 通过多头注意力机制提高了 KAP 网络模型的并行计算能力, 同时更好的表示上下文信息. 因此, 通过 BERT 预训练 KAP 网络模型能够更好地获取文本向量和词与词之间的关系.

我们使用第 3.2 节方法生成的关键词信息, 将关键词信息 K_t={K_1, K_2, ⋯, K_n}依次输入的网络的关键词信息模块中, 该模型通过式 (8) 使用 LSTM 依次接受每个单词的词向量和上个时刻的隐藏层状态, 得到当前时刻的隐藏层状态.

$$S_{key\_t} = LSTM(K_{t-1}, S_{key\_t-1}) \tag{8}$$

其中, S_{key_t} 是当前关键词信息的输出信息, S_{key_t-1} 为上一时刻的关键词信息输出信息, 利用当前关键词的输出信息与编码器的隐藏层状态进行注意力得分的计算, 该注意力可以表示当前关键词对原文单词的关注程度, 使用的公式为:

$$e^t_{key\_i} = V^{\mathrm{T}} \tanh(W_h h_i + W_s S_{key\_t} + b_{\mathrm{attn}}) \tag{9}$$

其中, V、W_h、W_s、b_{attn} 为可学习参数, h_i 为编码器的隐藏层状态. 每个时间步长得到的 $e^t_{key_i}$和指针生成网络得到的 e^t_i 进行相加, 公式如式 (10):

$$e^t_s = e^t_i + e^t_{key\_i} \tag{10}$$

得到的 e^t_s 融合了关键词的信息, 能够引导注意力机制关注原文的主要内容. 最后模型使用 e^t_s 进行注意力的计算, 公式为:

$$a^t = Softmax(e^t_s) \tag{11}$$

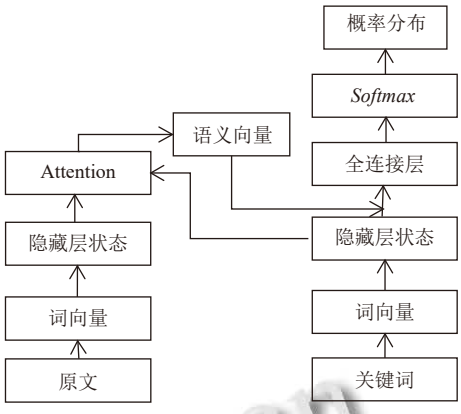

图 4 关键词提取流程图

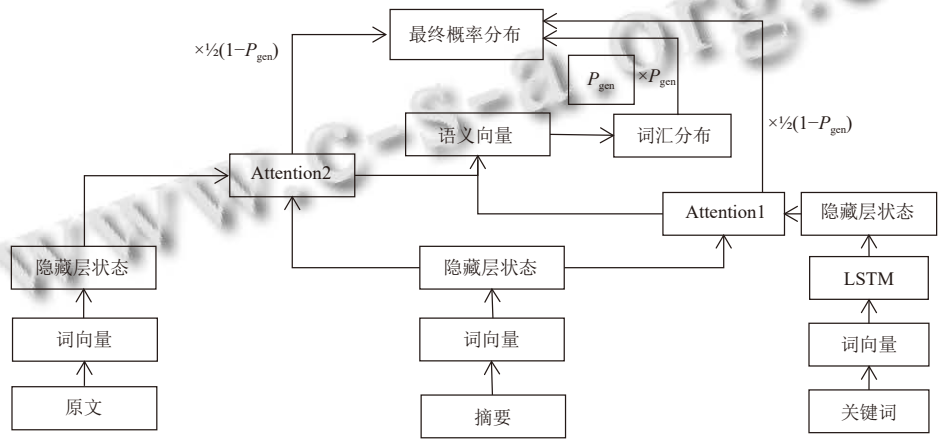

图 5 KAP 网络流程图

使注意力机制更能关注原文的关键信息, 达到关键词引导网络的效果, 以提高模型生成摘要的质量.

3.4 双指针

指针生成网络使用单指针进行单词的复制, 在 KAP 网络模型中, 单指针不能充分利用关键词信息, 为了实现从关键词信息中进行单词的复制, 我们使用双指针代替指针生成网络的单指针.

在原指针基础上, 新增一个拷贝关键词信息的指

针, 首先通过式 (9) 得到的 $e^t_{key\_i}$ 计算注意力得分 a^t_{k-1}.

$$a^t_{k\_i} = Softmax(e^t_{key\_i}) \tag{12}$$

通过得到在关键词上的注意力机制来计算在关键词上的拷贝概率.

$$P_{c\_k}(w) = \sum_{i:k_i=w} a^t_{k\_i} \tag{13}$$

其中, $P_{c\_k}(w)$ 是在关键词集合复制单词的概率, 最终预测单词分布的公式为:

$$P(w) = P_{\text{gen}}P_{\text{vocab}}(w) + \frac{1}{2}(1-p_{\text{gen}})(p_{c\_s}(w)+p_{ck}(w)) \tag{14}$$

其中, $P_{c\_s}(w)$ 是在原文上复制单词的概率, $P_{\text{vocab}}(w)$ 是生成单词的概率. 最终 KAP 网络模型通过 P_{gen} 来判断是在词汇表上生成单词还是从原文和关键词上复制单词, 达到从关键词和原文中复制单词的目的, 提高指针使用效率.

在训练过程中, 时间步长 t 的损失是目标词w^*_t的对数损失, 公式为:

$$loss_t = -\log(w^*_t) \tag{15}$$

则输入序列的整体损失为:

$$loss = \frac{1}{T}\sum_{t=0}^{T} loss_t \tag{16}$$

4 实验分析

4.1 数据集

实验数据来源于 LCSTS 数据集[7]. 该数据集是哈工大在新浪新闻采集得到的, 该数据集包含了超过 200 万条新闻摘要数据对. 数据集的具体情况如表 1 所示. PART I 的数据是没有被进行人工标注的, 而 PART II 和 PART III 的数据是被人工进行打分的. 分数从 1 到 5, 分数高低表明短文本与参考摘要之间的相关性.

表 1 LCSTS 数据集

评分情况	PART I	PART II	PART III
评分1	—	942	165
评分2	—	1 039	216
评分3	—	2 019	227
评分4	—	3 128	301
评分5	—	3 538	197
数据量	2 400 591	10 660	1 106

在实验过程中, 我们使用 PART I 作为训练集, PART II 和 PART III 分别作为验证集和测试集.

4.2 实验环境及参数设置

本实验在 Ubuntu 下进行, 实验算法使用 Python 3.6、TensorFlow 2.2.0 、Torch 1.5.0 进行实验, 实验参数设置如表 2 所示. 实验使用 Adam 算法[15] 进行模型的优化, 模型每隔 200 步保存一次, 直到达到最优结果.

表 2 实验参数设置

名称	参数
BERT-base模型层数	12
BERT隐藏层维度	768
BERT模型batch_size	16
学习率	5E–3
KAP模型词表大小	50k
KAP隐藏层维度	512
KAP模型batch_size	8
文本最大长度	100
摘要最大长度	20
抽取关键词个数	6

4.3 评价指标

本文使用 Lin[16] 提出的 ROUGE 指数进行模型的评估, 该 ROUGE 得分是计算模型生成的摘要与标准摘要的重叠数目来评价生成摘要的质量. 其中 *ROUGE-N* 计算公式为:

$$ROUGE\text{-}N = \frac{\sum_{S\in\{ReferenceSummaries\}}\sum_{gram_n\in S} Count_{\text{match}}(gram_n)}{\sum_{S\in\{ReferenceSummaries\}}\sum_{gram_n\in S} Count(gram_n)} \tag{17}$$

N 的值经常采用 1 和 2, 主要统计在 N-gram 上的召回率, 反映了生成摘要的准确性. ROUGE-L 是最长公共子序列的重合率计算, 计算公式如下:

$$R_{\text{LCS}} = \frac{LCS(S,C)}{len(C)} \tag{18}$$

$$P_{\text{LCS}} = \frac{LCS(S,C)}{len(S)} \tag{19}$$

$$F_{\text{LCS}} = \frac{(1+\beta^2)R_{\text{LCS}}P_{\text{LCS}}}{R_{\text{LCS}}+\beta^2 P_{\text{LCS}}} \tag{20}$$

其中, S 表示模型生成摘要, C 是原文摘要, R_{LCS} 表示召回率, P_{LCS} 表示精确率, F_{LCS} 为 ROUGE-L 得分. ROUGE-L 使用最长公共子序列衡量, 在一定程度上能够体现生成摘要的可读性.

4.4 关键词抽取结果

表 3 是关键词抽取结果, 该结果是由序列到序列

模型和 TextRank 两部分组成, 由于序列到序列模型能够生成词汇表上的单词, 在一定情况可以生成新颖的关键词.

表 3 关键词抽取结果

新闻原文	关键词
农夫山泉在出具了4个水源地4个批次成品水检测报告后, 又出具了一份美国国家测试实验室对其成品水的检测报告. 对此, 有网友称: 农夫山泉, 你别再兜圈子了, 产品品质高于美国标准? 高于月球标准都没有用, 执行低要求地方标准是你的硬伤! (京华时报)	农夫山泉 祭奠 标准 成品 美国 测试
目前, 金融改革正处在难得的战略机遇期, 这是由国内外经济形势决定的. 分析人士指出, 目前进一步推进改革的条件要好于以往, 改革可以向前推进、向下深化. 同时, 金融的配套设施建设和改革必须引起足够重视.	金融 改革 启航 形势 分析 机遇期

4.5 实验结果比较

将上述提取的关键词融入到指针生成网络中, 表 4 列出了 KAP 网络模型以及其他文本摘要模型在 LCSTS 数据集上的实验结果. 为了验证本文方法在摘要生成中的有效性, 我们选取了一些具有代表性的算法和指针生成网络进行比较.

表 4 ROUGE 评价结果对比

模型	ROUGE-1	ROUGE-2	ROUGE-L
RNN	21.5	8.9	18.6
RNN-context	29.9	17.4	27.2
CopyNet	34.4	21.6	31.3
SRB	33.3	20.0	30.1
HAM	35.4	11.9	33.3
DRGD	36.99	24.15	34.21
KIGN	37.76	16.56	34.49
PGN	36.24	19.16	32.90
BERT-PGN	37.78	20.61	34.30
KAPO	38.91	21.56	35.54
KAP	**39.25**	**22.31**	**35.93**

(1) RNN[5] 是基于 RNN 的 Seq2Seq 模型, 将编码器最后一个隐藏层作为解码器的输入, 没有加注意力机制.

(2) RNN-context[5] 是基于 RNN 带有注意力机制的 Seq2Seq 模型.

(3) CopyNet[6] 是基于注意力的具有复制机制的 Seq2Seq 模型, 在解码器阶段添加了生成和复制模式, 允许从源文本进行内容的复制.

(4) SRB[17] 是一种提高源文本和摘要之间语义相关性的模型, 通过引入了一个基于语义关联的神经模型来提高文本和摘要之间的语义的相似度.

(5) HAM[18] 方法首先是要自我注意力机制发现原句之间的关系, 之后将复制机制加入到网络中.

(6) DRGD[19] 是采用递归隐随机模型学习目标摘要中隐含的结构信息的深度循环模型.

(7) KIGN[13] 是一种将抽取法和抽象法相结合的摘要生成模型, 该模型是关键词编码引导摘要进行生成.

(8) PGN[8] 是我们使用的基线模型: 指针生成网络.

(9) BERT-PGN 是基于 BERT 预训练的指针生成网络.

KAPO 是在单指针生成网络上加入关键词信息. KAP 是实验的最终模型, 在指针生成网络上融入关键词信息和加上双指针拷贝机制. 从表 3 看出, 最终模型与 RNN 和 RNN-context 相比都有很大程度的提升, 与 CopyNet、SRB、HAM 等算法比较有一定的提升. 在基线模型 PGN 比较上, 融入关键词信息的指针生成网络对比基线模型在 ROUGE-1 得分上高了 2.67, 在 ROUGE-2 得分上高了 2.4, 在 ROUGE-L 高了 2.64. 融入关键词信息的指针生成网络和加入双指针拷贝机制对比基线模型在 ROUGE-1 得分上高了 3.01, 在 ROUGE-2 得分上高了 3.15, 在 ROUGE-L 高了 3.03, 与 BERT-PGN 相比都有一定的提高. 结果表明, 与没有关键词信息的网络相比, 我们的结果得到了改进, 这也表明该模型对我们提取的关键字信息进行了学习.

图 6 折线图是表 3 各种模型 ROUGE 分数对比的另一种直观展示, KAP 我们的最终模型. 可以看出我们的模型在 ROUGE-2 和 ROUGE-3 的得分是最高的.

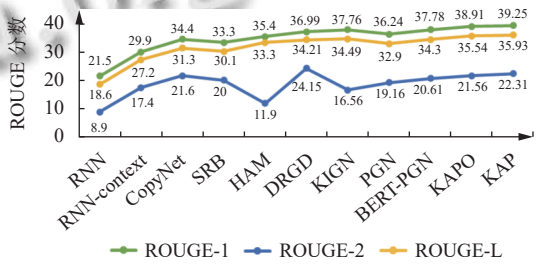

图 6 各种模型的 ROUGE 分数对比

我们对比 KAP 的单指针和双指针, 对比如图 7 所示. 从图中可以看出双指针下模型在 ROUGE 得分上都有一定的提高, 具体得到在 ROUGE-1 得分上高了 0.34, 在 ROUGE-2 得分上高了 0.75, 在 ROUGE-L 得分上高了 0.39. 从数据看出将双指针代替原来的单指针是必要的.

在表 5 我们列出了两个摘要生成的案例, 其中 PGN

是我们的基线模型指针生成网络, 我们的模型是融入关键词信息和加入双指针拷贝机制. 从表中可以看出, 第 1 个示例我们的模型比基线模型多了“环比”一词, 第 2 个示例我们生成的摘要包含更多的原文信息. 可以看出 PGN 生成的内容不全面, 只生成摘要的一部分或者丢失信息. 在融入关键词信息后, 生成的摘要更加契合原文内容, 能够包含更多的原文信息.

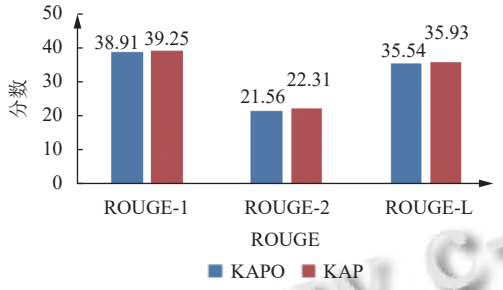

图 7 KAP 单双指针对比

表 5 摘要生成示例

序号	模型	内容
1	新闻原文	数据显示, 6月全国百城新建住宅均价为10 258元/平, 环比上涨0.77%, 已连续13个月环比上涨, 广州上涨2.74%, 居全国第五. 十大城市住宅均价为17 376元/平. 在需求推动、土地市场高温等众多因素的推动下, 下半年房价上涨压力依然较大.
	参考摘要	百城住宅均价连续13个月环比上涨
	PGN	6月百城房价连涨13个月
	本文	6月百城房价环比连涨13个月
2	新闻原文	28日晚, @AFU阿芙创始人孟醒连发几条微博, 指责京东商城拖欠其100万货款长达13个月, 要求京东马上结清货款. 京东商城回应, 货款未结算在于涉及的增值税发票信息出现错误. 阿芙方面暂未对此说法置评.
	参考摘要	京东回应阿芙欠款事件称结款延迟源于发票错误
	PGN	京东商城回应欠款门报道
	本文	京东回应欠款发票信息出现错误

为了进一步突出模型的可读性, 我们加入了李克特考量表来对文本摘要结果进行人工评估. 我们从最终生成的摘要中随机选择了 200 个摘要, 并邀请 12 名研究生对所选摘要进行评分. 分数从 1 到 5, 分数越高表明生成摘要的可读性越好. 从表 6 中可以看出, 得分为 1 的有 29 个, 得分为 2 的有 47 个, 得分为 3 的有 57 个, 得分为 4 的有 36 个, 得分为 5 的 31 个. 可以得出, 得分高于 3 分有 67 个句子, 占 33.5%, 得分高于 2 的有 124 个句子, 占 62%. 因此, 从分析的数据可以得出结论, 所生成摘要的有一定的可读性的.

表 6 李克特调查语义摘要的可读性 (200 个摘要)

人工打分	数量
1	29
2	47
3	57
4	36
5	31

5 结论与展望

本文关键词引导指针生成网络进行摘要生成. 首先我们使用提取模型从输入文本获取关键信息. 然后, 将关键信息编码到指针生成网络的注意力机制中, 引导摘要的生成. 此外, 使用的双指针拷贝机制, 在输入的文本和关键词中进行单词的复制, 扩大了单词的复制范围. 实验表明, 我们的模型在 ROUGE 得分有明显的提高, 同时生成的摘要能够契合原文内容. 在未来, 我们可以改进模型的骨架, 通过使用 Transformer 能更好地捕获词与词之间的关系, 进一步提高模型的准确性和关联性.

参考文献

1 Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal: ACM, 2014. 3104–3112.
2 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: ACL, 2015. 379–389.
3 Over P, Dang H, Harman D. DUC in context. Information Processing & Management, 2007, 43(6): 1506–1520.
4 Graff D, Kong JB, Chen K, *et al*. English gigaword. Linguistic Data Consortium, Philadelphia, 2003, 4(1): 34.
5 Chopra S, Auli M, Rush AM. Abstractive sentence summarization with attentive recurrent neural networks. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego: Association for Computational Linguistics, 2016. 93–98.
6 Gu JT, Lu ZD, Li H, *et al*. Incorporating copying mechanism in sequence-to-sequence learning. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: ACL, 2016. 1631–1640.
7 Hu BT, Chen QC, Zhu FZ. LCSTS: A large scale Chinese short text summarization dataset. Proceedings of the 2015

Conference on Empirical Methods in Natural Language Processing. Lisbon: Association for Computational Linguistics, 2015. 1967–1972.

8 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: ACL, 2017. 1073–1083.

9 Mihalcea R, Tarau P. TextRank: Bringing order into text. Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing. Barcelona: ACL, 2004. 404–411.

10 巴志超, 李纲, 朱世伟. 共现分析中的关键词选择与语义度量方法研究. 情报学报, 2016, 35(2): 197–207. [doi: 10.3772/j.issn.1000-0135.2016.002.009]

11 Wan XJ, Yang JW, Xiao JG. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics. Prague: ACL, 2007. 552–559.

12 Li CL, Xu WR, Li S, *et al*. Guiding generation for abstractive text summarization based on key information guide network. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics. New Orleans: ACL, 2018. 55–60.

13 Li HR, Zhu JN, Zhang JJ, *et al*. Keywords-guided abstractive sentence summarization. Proceedings of the 34th AAAI Conference on Artificial Intelligence. New York: AAAI, 2020. 8196–8203.

14 Devlin J, Chang MW, Lee K, *et al*. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Minneapolis: ACL, 2019. 4171–4186.

15 Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv: 1412.6980, 2014.

16 Lin CY. ROUGE: A package for automatic evaluation of summaries. Proceedings of Workshop on Text Summarization Branches Out. Barcelona: ACL, 2004. 74–81.

17 Ma SM, Sun X, Xu JJ, *et al*. Improving semantic relevance for sequence-to-sequence learning of Chinese social media text summarization. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017. 635–640.

18 Yang WJ, Tang ZC, Tang XH. A hierarchical neural abstractive summarization with self-attention mechanism. Proceedings of the 2018 3rd International Conference on Automation, Mechanical Control and Computational Engineering. Atlantis Press, 2018. 514–518.

19 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: ACL, 2017. 2091–2100.

(校对责编: 牛欣悦)