软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*,2019,30(9):2705−2717 [doi: 10.13328/j.cnki.jos.005850]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

# 全局自匹配机制的短文本摘要生成方法*

吴仁守, 王红玲, 王中卿, 周国栋

(苏州大学 计算机科学与技术学院,江苏 苏州 215006)

通讯作者: 王红玲, E-mail: hlwang@suda.edu.cn

**摘 要**: 基于编码器-解码器架构的序列到序列学习模型是近年来主流的生成式自动文摘模型,其在计算每一个词的隐层表示时,通常仅考虑该词之前(或之后)的一些词,无法获取全局信息,从而进行全局优化.针对这个问题,在编码器端引入全局自匹配机制进行全局优化,并利用全局门控单元抽取出文本的核心内容.全局自匹配机制根据文本中每个单词语义和文本整体语义的匹配程度,动态地从整篇文本中为文中每一个词收集与该词相关的信息,并进一步将该词及其匹配的信息有效编码到最终的隐层表示中,以获得包含全局信息的隐层表示.同时,考虑到为每一个词融入全局信息可能会造成冗余,引入了全局门控单元,根据自匹配层获得的全局信息对流入解码端的信息流进行过滤,筛选出原文本的核心内容.实验结果显示,与目前主流的生成式文摘方法相比,该方法在 Rouge 评价上有显著提高,这表明所提出的模型能有效融合全局信息,挖掘出原文本的核心内容.

**关键词**: 自匹配机制;全局信息;神经网络;自动文摘;自然语言生成

**中图法分类号**: TP18

中文引用格式: 吴仁守,王红玲,王中卿,周国栋.全局自匹配机制的短文本摘要生成方法.软件学报,2019,30(9):2705−2717. http://www.jos.org.cn/1000-9825/5850.htm

英文引用格式: Wu RS, Wang HL, Wang ZQ, Zhou GD. Short text summary generation with global self-matching mechanism. Ruan Jian Xue Bao/Journal of Software, 2019,30(9):2705−2717 (in Chinese). http://www.jos.org.cn/1000-9825/5850.htm

## Short Text Summary Generation with Global Self-matching Mechanism

WU Ren-Shou, WANG Hong-Ling, WANG Zhong-Qing, ZHOU Guo-Dong

(School of Computer Science and Technology, Soochow University, Suzhou 215006, China)

**Abstract**: In recent years, the sequence-to-sequence learning model with the encoder-decoder architecture has become the mainstream summarization generation approach. Currently, the model usually only considers limited words before (or after) when calculating the hidden layer state of a word, but can not obtain global information, so as to optimize the global situation. In order to address above challenges, this study introduces a global self-matching mechanism to optimize the encoder globally, and proposes a global gating unit to extract the core content of the text. The global self-matching mechanism dynamically collects relevant information from the entire input text for each word in the text according to the matching degree of each word semantics and the overall semantics of the text, and then effectively encodes the word and its matching information into the final hidden layer representation to obtain the hidden layer representation containing the global information. Meanwhile, considering that integrating global information into each word may cause redundancy, this study introduces a global gating unit, filters the information flow into the decoder according to the global information obtained from the self-matching layer, and filters out the core content of the source text. Experimented result shows that the proposed model has a significant improvement in the Rouge evaluation over the state-of-the art method.

**Key words**: self-matching mechanism; global information; neural networks; automatic text summarization; natural language generation

* 基金项目: 国家自然科学基金(61806137); 江苏省高等学校自然科学研究(18KJB520043); 江苏高校优势学科建设工程

Foundation item: National Natural Science Foundation of China (61806137); Jiangsu High School Research (18KJB520043); A Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions

收稿时间: 2019-01-07; 修改时间: 2019-03-02; 采用时间: 2019-04-04; jos 在线出版时间: 2019-05-22

CNKI 网络优先出版: 2019-05-22 15:26:15, http://kns.cnki.net/kcms/detail/11.2560.TP.20190522.1525.009.html

自动文摘是自然语言处理领域的一个重要研究方向,其目的是通过对原文本进行压缩、提炼,为用户提供能够覆盖原文核心内容且简明扼要的文字描述.自动文摘应用最广范的是在新闻领域,这是由于新闻信息的过载,人们迫切地希望有一种工具可以帮助他们在尽可能短的时间内了解更多有价值的新闻.此外,随着互联网上发布的数据日益增多,搜索引擎也成为其应用之一,例如,基于查询(query-based)的自动文摘可以帮助用户尽快找到感兴趣的内容.

自动文摘从所采用的实现方法上考虑,可以分为抽取式文摘(extractive summarization)和生成式文摘(abstractive summarization).抽取式文摘从原始文档中直接抽取重要性最高的若干个结构单元(句子、段落等)组成摘要,通常该方法比较简单易行,目前已经发展较为成熟.不过,抽取式文摘也存在一些固有的缺陷,例如不能确保摘要的连贯性和衔接性.相对而言,生成式文摘通常利用自然语言理解技术对原文档进行语法语义的分析,获取原文档的主要内容,然后通过语言模型、信息融合、信息压缩等自然语言生成技术生成摘要.该方法的优点是文摘结果跳出了原文档句子的局限,其摘要句不局限于原文档中的句子,能够较好地处理冗余,强调主题连贯性;缺点是生成的句子可读性差,只能在一定程度上确保上下文的连贯性和衔接性.

近年来,深度学习被广泛应用于自然语言处理任务并取得了一些成果.特别地,基于编码器-解码器(encoder-decoder)架构的序列到序列学习模型(sequence-to-sequence learning,简称 seq2seq)能够比较有效地将文本从一种形式转换为另一种形式,例如机器翻译[1]和语音识别[2].由于深度学习具有较强的泛化能力,可以学习到文本的隐含特征,避免繁琐的人工特征提取,实现了传统自动文摘系统中关键的重要性评估、内容选择等模块端到端一体化,相关方法在文摘任务上的应用研究受到了广泛关注.不过,这些方法往往需要规模远大于传统方法的训练语料,加上当前主流的神经网络框架尚不能够有效对长文档进行语义编码,因此目前的相关研究大多集中于短文本的摘要生成[3].该任务通常仅以文档首句作为输入,以一个短句作为输出(见表 1).本文也将针对短文本进行摘要生成研究.

由于缺少原始文档和摘要之间的短语对齐,自动文摘任务比语言之间的翻译困难得多.自动文摘任务要求系统全面准确地理解文档所表达的意思,然后用可读性强的人类语言将其简练地总结出来.因此,完整的全局信息对于自动文摘系统全面准确地获取文档的主要内容至关重要.虽然在解码阶段的每个时间步中,已有的模型大多已采用注意力机制对编码器的输入序列进行加权求和,以获得原始文档的全局信息,但是在编码阶段,传统的编码器在计算每一个词的向量表示或者隐层状态时仅考虑该词之前(或之后)的一些词,而不是完整的全局信息.另外,采用双向循环神经网络(RNN)[4]得到的前后向信息仅进行简单拼接,无法有效融合相关信息,导致了次优化,生成的摘要往往会缺失或偏离原文档核心信息.例如,在表 1 中,基于双向循环神经网络编码器的 seq2seq 模型生成的摘要就忽略了“二氧化硫”的来源问题.

**Table 1** An sample of the short text summary

**表 1** 短文本摘要示例

| |
|---|
| 文本:北京五环内很少燃煤,为什么中午二氧化硫会出现峰值?“本次污染事件突发的一个重要原因,是以河北燃煤排放为主的二氧化硫一夜之间转化成硫酸盐.”中科院大气物理研究所研究员王跃思称,“北京城区白天空气中二氧化硫出现高值,大部分来自外地输送.” |
| 参考摘要:中科院:北京空气中的二氧化硫来自河北燃煤 |
| seq2seq:中科院专家:北京雾霾天气中二氧化硫出现高值 |

针对上述问题,本文提出了一种全局自匹配机制,通过自匹配来自编码器的信息流,将上下文中的全局信息整合到原始文本每个词的表示中.首先,用编码器对输入文本进行编码;然后,利用配备有自匹配机制的自匹配层动态地对编码后的输入文本进行自匹配.具体而言,对于原始文本中的每个单词,全局自匹配机制根据整个输入文本中每个单词语义和文本整体语义的匹配程度,动态地从整个输入文本中收集与该单词相关的信息,并将单词表示和相关信息融合到最终隐层表示中.同时,考虑到为每一个词收集与该单词相关的全局信息可能会造成信息的冗余,本文引入了全局门控单元对自匹配层获得的包含上下文信息的隐层表示进一步筛选,去除冗余信息,以便挖掘出原文本的核心内容并用于解码器生成摘要.

综上所述,本文提出了一个基于编码器-解码器架构的生成式自动文摘模型,该模型由编码器、全局自匹配

层、全局门控单元和基于注意力机制的解码器组成,并在 LCSTS[5]数据集上对该模型进行了系统深入的实验,实验结果表明,具有全局自匹配机制和全局门控单元的模型能够生成具有较高准确性的摘要,并且在连贯性和衔接性方面始终优于不使用自匹配机制和全局门控单元的模型.

## 1 相关工作

传统的自动文摘主要为抽取式文摘,因抽取式文摘文献众多,在此不再赘述,仅给出一些经典的方法,具体包括:(1) 基于统计模型的方法,如 Chali[6]提出的一种基于 SVM 的多文本自动文摘方法,使用一组 SVM 分类器,充分利用 SVM 的泛化特性,抽取出最能代表文档核心内容的句子;(2) 基于聚类的方法,主要利用多文档集合的信息,将多文档集合作为一个整理进行研究,测量所有句子对之间的相似性,在此基础上,用各种聚类方法(*K*-Means,*K*-Medoids,AP 等)识别公共信息的主题,并从每个类别中抽取中心句子作为文档摘要,如 Siddharthan 等人[7]的工作;(3) 基于图模型的方法,如 Mihalcea 和 Tarau 提出的 TextRank[8]算法,将句子间的相似关系看成了一种推荐或投票关系,并构建了 TextRank 网络图,通过迭代计算至收敛来得到句子的权值.

近几年来,深度神经网络模型因其强大的表征能力,在分布式语义[9]、语言模型[10]、机器翻译[1]等领域不断推进机器智能的极限.类似地,目前生成式自动文摘也主要依靠基于编码器-解码器架构的序列到序列学习模型,其中,编码器、解码器均由数层循环神经网络构成,编码器负责把原文编码为语义向量 *C*;解码器负责从这个语义向量 *C* 中提取信息,获取语义,生成文本摘要.但是由于长距离依赖问题的存在,RNN 到最后一个时间步输入单词的时候,已经丢失了相当一部分信息.这时候,编码生成的语义向量 *C* 同样也丢失了大量信息,导致生成的摘要不够准确.为了解决这一问题,Rush 等人[11]首次将应用于机器翻译任务中的注意力机制(attention mechanism)[12]引入自动文摘任务中,并在相关数据集上取得了良好的效果.注意力机制是一种注意力(资源)分配机制,在某个特定时刻,它总是重点关注与该时刻相关的内容,其他内容则进行选择性忽视.例如在翻译“Knowledge”时只会关注“知识”,这样的对齐能让文本翻译或者摘要生成更具针对性.

作为 Rush 等人[11]工作的扩展,Chopra 等人[13]使用类似的卷积模型作为编码器,同时把解码器换成了 RNN,在同样的数据集上产生了性能更好的结果.Li 等人[14]在基于注意力机制的序列到序列模型的基础上增加了潜在结构向量,来学习目标摘要中隐含的潜在结构信息,以提高摘要质量.另外,Zeng 等人[15]提出了一种重读机制,在编码器计算每一个词的表示之前先阅读一遍输入序列,然后利用第 1 次读取得到的隐层状态帮助第 2 次读取文本时的表示生成.该想法的基本动机是:考虑到人们在阅读一篇文章时,通常需要读完一遍才能去确认文中哪些词是重点.为了解决未登录词问题(out of vocabulary,简称 OOV),Gu 等人[16]提出了一种复制机制来复制输入序列的适当片段并将其放入输出序列.See 等人[17]提出了一种混合指针生成器网络,在保留生成器产生新词的能力的同时,可以从原文本复制单词,大幅度提高了信息再现的准确性;同时,新增了一个覆盖机制以防止重复.同样的,针对解码时可能会不断重复已有的单词,Lin 等人[18]提出了全局编码框架来尝试解决这个问题.该框架由卷积门控单元组成,用于执行全局编码,以改善源端信息的表示.Ma 等人[19]引入了基于语义相关性的神经模型,来鼓励文本和摘要之间的高语义相似性.

在利用输入文档全局信息进行摘要生成方面,目前主要有 Zeng 等人[15]和 Lin 等人[18]的工作,两者分别从不同角度,利用输入文档的全局信息来指导摘要生成.本文方法同样使用了全局信息,但是使用方法有所不同,与他们的主要区别如下所示.

(1) Zeng 等人[15]首先使用编码器通读一次输入文本,将得到的第 1 以及最后一个时间步的隐层状态作为整篇文档的特征向量,来计算每个单词的重要性权重向量用于第 2 次阅读.不同于 Zeng 等人[15]对于文中的每一个词都使用了相同的全局特征向量,考虑到文中不同词关注的全局信息应该是有所不同的,本文提出的方法动态地从整个输入文本中为文本中每一个词收集与该词相关的信息作为其对应的全局特征向量;

(2) Lin 等人[18]提出了一种全局编码框架,根据源端上下文的全局信息控制从编码器到解码器的信息流,其由卷积门控单元组成,用于执行全局编码以改进源端信息的表示.虽然其利用了全局信息,但是它

仅仅利用门控单元对源端的输入信息进行筛选,而没有将源端信息和全局信息进行有效的融合.本文在获取全局信息后,将源端每个词的表示与其对应的全局信息进行了融合,用于解码器生成摘要.

## 2 基于全局自匹配机制的短文本摘要生成方法

给定输入文档 $D$,将其表示为单词序列 $D=(w_1,w_2,...,w_{T_d})$,其中,每个单词 $w_i$ 来自固定的词汇表 $V$.自动文摘旨在将 $D$ 作为输入,并生成简短的摘要 $y=(y_1,y_2,...,y_{T_y})$,其中,$T$ 表示序列长度,输入文档序列长度 $T_d$ 大于生成摘要序列长度 $T_y$.

本文提出的基于全局自匹配机制的短文本摘要生成方法的系统框架如图 1 所示,主要包括基于双向长短期记忆单元(long short-term memory,简称 LSTM)[20]的编码器、全局自匹配层、全局门控单元和配备注意力机制的长短期记忆单元解码器.其中,编码器读取输入文档,并构建其表示;全局自匹配层对编码后的输入文本进行自匹配,将全局信息融入输入文本表示中;全局门控单元对这些表示进行进一步筛选,并将其提供给解码器;解码器负责摘要生成.下面,将分别介绍编码器、全局自匹配层、全局门控单元和解码器的细节及其训练方法.

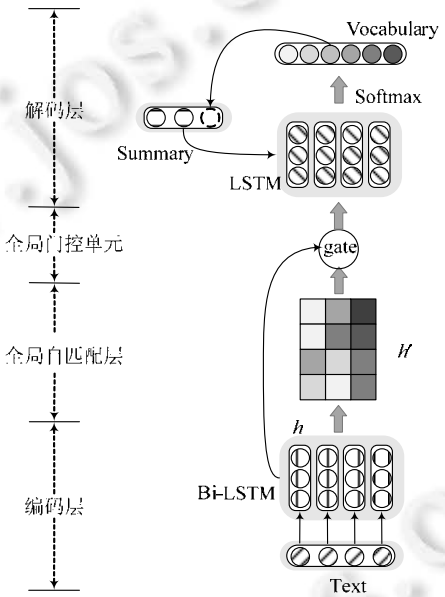


Fig.1 Model architecture overview

图 1 系统框架图

### 2.1 双向LSTM编码器

给定输入文档 $D=(w_1,w_2,...,w_{T_d})$,使用词嵌入矩阵 $W_e$ 将输入文档中的词 $w_t,t\in[0,T_d]$ 转换为连续表示 $x_t$,具体见公式(1):

$$x_t=W_e w_t,t\in[0,T_d] \tag{1}$$

在获得文档连续表示之后,利用双向循环网络对输入序列进行编码[21].双向循环网络由前向和后向循环网络组成:前向循环网络正向读取输入序列(从 $x_1$ 到 $x_{T_x}$),并计算前向隐藏层状态 $(\vec{h}_1,\vec{h}_2,...,\vec{h}_{T_x})$;而后向循环网络从反向读取输入序列(从 $x_{T_x}$ 到 $x_1$),并计算反向隐藏状态 $(\overleftarrow{h}_{T_x},\overleftarrow{h}_{T_x-1},...,\overleftarrow{h}_1)$.对于每个单词 $x_t$,将它对应的前向隐藏状态向量 $\vec{h}_t$ 和后向隐藏状态向量 $\overleftarrow{h}_t$ 拼接起来表示 $x_t$ 对应的隐藏层状态表示 $h_t$.

具体计算方法见公式(2)~公式(4):

$$\vec{h}_t=\vec{f}(x_t,\vec{h}_{t-1}) \tag{2}$$

$$\overleftarrow{h}_t=\overleftarrow{f}(x_t,\overleftarrow{h}_{t+1}) \tag{3}$$

$$h_t=[\vec{h}_t;\overleftarrow{h}_t] \tag{4}$$

其中,$h_t\in\mathbb{R}^n$ 是 $t$ 时刻的隐藏层状态.$f(\cdot)$是一些非线性函数,在经过对模型性能和训练复杂性之间进行权衡之后,我们选择了长短期记忆单元.

### 2.2 全局自匹配层

与其他自然语言生成任务相比,例如机器翻译,自动文摘更注重获取原文档的主要内容来生成摘要.传统的基于单向循环神经网络的编码器在计算输入文档中每一个词对应的隐藏层状态时,仅仅考虑了在该词之前的一些词,并不是完整的上下文信息.虽然上述基于双向 LSTM 的编码器在计算输入文档中每一个词对应的隐藏层状态时分别考虑到了该词之前和之后的一些词,但是得到的前、后向信息还是局部的(包含之前部分或之后部分),仅仅做了拼接,没有进行有效地融合,无法得到针对每一个词特定的全局文档信息.因此,在利用编码器获得输入文档的隐藏层状态之后,我们希望将全局文档信息纳入文档中每个词对应的隐藏层状态中来增强原有的隐藏层状态,弥补上述的不足.

在过去关于句子对表示的研究中,Rocktäschel 等人[22]提出通过对句子对中的单词进行软对齐来生成句子对表示.在机器阅读理解任务中,Wang 和 Jiang[23]介绍了一种 Match-LSTM 单元,它在传统 LSTM 单元的基础上,将文章表示作为循环网络每一次输入的附加输入来指导编码,从而使输出的每个隐藏层状态包含全局的文章信息.为确定文章中各个部分的重要性并获取与问题相关的部分,Wang 等人[24]在 match-LSTM 的基础上又添加了一个选择门来控制循环网络的输入,该门有效地模拟了在阅读理解任务中只有部分文本与问题相关的现象.

受上述参考工作的启发,本文首次将匹配机制引入自动文摘任务,并针对自动文摘任务的特性进行改进.在传统的阅读理解任务中,匹配机制主要用于计算文章中每个单词语义和问题整体语义的匹配程度,以凸显出哪些单词是问题答案的可能性.根据自动文摘任务更注重获取原文档主要内容的特性,我们提出了一种全局自匹配机制来对输入文档进行自身到自身的匹配.与 Wang 等人[24]提出的自匹配注意力(self-matching attention)不同,其针对阅读理解任务的特性,在自匹配过程中通过一个选择门对文章中与问题相关的部分进行筛选,确定答案可能存在的位置.本文提出的全局自匹配机制根据文档中每个单词语义和原文档整体语义的匹配程度,动态地从整个原文档中为文中每一个词收集与该词相关的信息后,进一步将该词及其匹配的信息进行融合,将其对应的全局信息编码到该词最终的隐层表示中,以获得包含全局信息的隐层表示.形象地,可以将输入文档成对表示,文档对中的两篇文档都为输入文档,将其中一篇文档视为机器阅读理解任务中的问题.自匹配机制根据问题动态确定文档各部分的重要性,为问题中的每个词获取其相关的部分来扩展该词对应的表示,从而获得对应的原始文档全局信息.本文提出的全局自匹配机制的具体结构如图 2 所示,下面将进行详细描述.

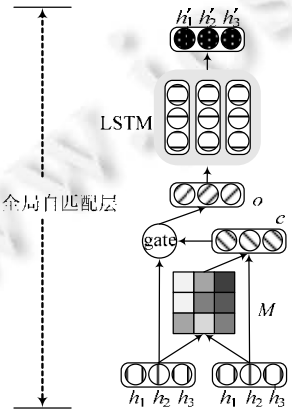


Fig.2 Structure of global self-matching layer

图 2 全局自匹配层结构图

首先,使用点积计算成对匹配矩阵 $M$,它表示问题中的每一个单词 $s$ 和输入文档中的每一个单词的成对匹

配程度,两者越相关,匹配程度越大.根据得到的成对匹配矩阵 $M$,可以为文档中的每一个单词 $x_t$ 计算其对应的上下文信息 $c_t$,并通过将单词 $x_t$ 对应的隐藏层状态 $h_t$ 和上下文信息 $c_t$ 进行融合,得到中间表示 $r_t$.然后,全局信息是 $o_t$ 隐藏层状态 $h_t$ 和 $r_t$ 的线性插值.门 $g_t$ 自适应地控制当前单词 $x_t$ 对应的全局信息 $o_t$ 是应该从 $h_t$ 直接复制还是应该通过更复杂的路径 $r_t$.极端情况下,当 $g_t$=0 时,$o_t$=$h_t$,此时全局信息 $o_t$ 直接为 $x_t$ 对应全局信息 $r_t$,不再考虑其对应的隐藏层状态 $h_t$.最后,利用 match-LSTM 将全局信息 $o_t$ 作为循环网络的附加输入获得最终包含全局信息的隐藏层状态 $h'_t$.

形式上,当给定输入文档的隐藏层状态向量表示 $h=(h_1,h_2,...,h_{T_x})$ 时,可以计算输入文档中每一个单词 $x_t$ 对应的最终隐藏层状态 $h'_t$:

$$h'_t=f(h'_{t-1},[h_t;o_t]) \tag{5}$$

$$r_t=\tanh(W_r[h_t;c_t]) \tag{6}$$

$$g_t=\sigma(W_g[h_t;c_t]) \tag{7}$$

$$o_t=g_t\times r_t+(1-g_t)\times h_t \tag{8}$$

其中,$f(\cdot)$为 LSTM 单元.$c_t$ 是单词 $x_t$ 对应的整篇文档基于注意力机制的向量表示,其可以被计算为输入文档中每个词对应的隐藏层状态向量表示的加权和,即:

$$c_t=\sum_{i=1}^{T_x}M(t,i)h_i \tag{9}$$

其中,$M(t,i)$是计算输入文档中单词 $x_t$ 对应的上下文信息 $c_t$ 时,单词 $x_i$ 对应隐藏层状态向量的权重,表示文档中的单词 $x_t$ 和另一单词 $x_i$ 的相关程度,具体计算过程如下:

$$M(t,i)=\frac{\exp(h_t^c\cdot(h_i^x)^T)}{\sum_{k=1}^{T_x}\exp(h_t^c\cdot(h_k^x)^T)} \tag{10}$$

### 2.3 全局门控单元

在获取全局信息之后,我们使用全局门控单元对这些表示进行进一步筛选,以便去除冗余信息,挖掘出原文档的核心内容.全局门控单元 $g_{global}$ 根据编码器每个时间步的信息与全局信息的关系筛选输入解码端的信息流,其在每个维度上的输出值介于 0 和 1 之间的向量.在此,我们使用 Vaswani 等人提出的缩放点积注意力(scaled dot- product attention)[25]来计算每一个时间步的信息与全局信息的关系.缩放点积注意力可以被描述为将一个查询(query)和一组键(key)-值(value)对映射到一个输出,其中,查询、键、值和输出都是向量.本文中,查询为编码器每个时间步的输出,键和值同为自匹配后获得的全局信息.首先,将 query 和每个 key 进行相似度计算得到权重;第 2 步,使用 *softmax* 函数对这些权重进行归一化;最后,将权重和相应的键值 value 进行加权求和,得到最后的 attention:

$$g_{global}=softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{11}$$

$$Q=W_Qh,K=W_Kh',V=W_Vh' \tag{12}$$

$$h''=h'\times\sigma(g_{global}) \tag{13}$$

其中,$W_Q$,$W_K$,$W_V$ 为可训练矩阵参数,$\sigma(\cdot)$为 *sigmoid* 函数.

### 2.4 基于注意力机制的单向LSTM解码器

我们利用基于注意力机制的单向 LSTM 解码器来读取输入单词,并逐字生成摘要.在每个时间步骤,解码器通过从词汇表的分布中采样来生成摘要中的词,直到采样到表示句子结尾的标记时结束.具体计算方法如下:

$$p_{gen}(y_i|y_i,\ldots,y_{i-1},x)=g(y_{i-1}.s_i,c_i) \tag{14}$$

$$s_i=f(s_{i-1},y_{i-1},c_i) \tag{15}$$

$$c_i=\sum_{j=1}^{T_x}\alpha_{ij}h'_j \tag{16}$$

$$\alpha_{ij}=\frac{\exp(e_{ij})}{\sum_{k=1}^{T_x}\exp(e_{ik})} \tag{17}$$

$$e_{ij}=a(s_{i-1},h'_j) \tag{18}$$

其中,$f(\cdot)$为 LSTM 单元;$g(\cdot)$是非线性、潜在的多层函数,输出 $y$ 的概率;$s_t$ 为 $t$ 时刻 RNN 的隐藏状态;$a(\cdot)$是前馈神经网络.

为了解决罕见和未知的单词,Gulcehre 等人[26]提出使用指向机制从原句复制罕见的单词.我们在系统中应用这种指向方法.当解码单词 $y_i$ 时,复制开关将当前解码器状态 $s_i$ 和上下文向量 $c_i$ 作为输入,并产生从源端输入文档复制单词的概率 $p$:

$$p_{copy}(y_i|y_i,\ldots,y_{i-1},x)=\sigma(Ws_i+Uc_i+b) \tag{19}$$

其中,$\sigma$为 *sigmoid* 函数,$W$,$U$ 和 $b$ 为可训练参数.

### 2.5 训练与推理

给定输入文档 $D$,我们的模型可以使用随机梯度下降进行端到端训练,通过最小化生成摘要的负对数似然来训练模型参数,其训练过程本质上是逐步最大化生成摘要中每个词的概率,具体形式如下所示:

$$L=-\sum_{t=1}^{T_y}\log p(y_t\mid y_{<t},c,x;\theta) \tag{20}$$

其中,$Y_{<t}=\{y_1,y_2,\ldots,y_{t-1}\}$,$\theta$表示可训练的模型参数.

由于模型使用导师驱动过程(teacher forcing)进行训练,在生成 $t$ 时刻的单词时,不是由 $t$–1 时刻生成的单词作为输入,而是输入实际摘要中的预期单词.然而在测试过程中,生成 $t$ 时刻单词时,输入的是 $t$–1 时刻生成的单词,这导致了训练和测试之间的脱节.为了克服这种问题,在训练期间,我们随机地输入生成的单词,而不总是预期的单词[27].具体的,在生成 $t$ 时刻的单词时,我们以 0.1 的概率选择 $t$–1 时刻生成的单词作为输入,0.9 的概率输入实际摘要中的预期单词.在测试期间,我们使用集束搜索(beam-search)[28]解码器,其每次生成一个输入字,根据模型计算得到的 $y_t$ 的分布概率扩展 $B$ 个最高概率序列.

## 3 实验与评价

### 3.1 实验设置

- 实验数据

LCSTS 是取自于新浪微博的大规模中文短文本摘要数据集[5],数据集中包含了超过 200 万真实的中文短文本数据和每个文本作者给出的摘要.同时,还手动标记了 10 666 个简短摘要与相应短文的相关性.数据集由 3 部分组成(见表 2).

**Table 2** statistics information of LCSTS dataset

**表 2** LCSTS 数据集统计信息

| | | |
|---|---|---|
| 第 I 部分 | 2 400 591 | |
| 第 II 部分 | 成对数 | 10 666 |
| | 人工评分:1 | 942 |
| | 人工评分:2 | 1 039 |
| | 人工评分:3 | 2 019 |
| | 人工评分:4 | 3 128 |
| | 人工评分:5 | 3 538 |
| 第 III 部分 | 成对数 | 11 06 |
| | 人工评分:1 | 165 |
| | 人工评分:2 | 216 |
| | 人工评分:3 | 227 |
| | 人工评分:4 | 301 |
| | 人工评分:5 | 197 |

第 I 部分是 LCSTS 的主要内容,包含 2 400 591(短文本,摘要)对.这些对可用于训练有监督学习模型以进行摘要生成;第 II 部分包含 10 666 个人工标注的(短文本,摘要)对,评分范围从 1~5,表示短文本和相应摘要之间的相关性:“1”表示“最不相关”,“5”表示“最相关”;第 III 部分包含 1 106 对.对于这一部分,由 3 个标注者标记相同的 2 000 文本,最后保留具有共同分数的文本.该部分独立于第 I 部分和第 II 部分.

在本文实验中,我们使用第 I 部分作为训练集,第 II 部分作为开发集,第 III 部分中得分为 3、4 和 5 的子集作为测试集.

- 评价方法

ROUGE[29]是 Lin 在 2004 年提出的一种自动摘要评价方法,被广泛应用于 NIST 组织的自动摘要评测任务中.ROUGE 基于摘要中 *n* 元词(*n*-gram)的共现信息来评价摘要,是一种面向 *n* 元词召回率的评价方法.基本思想为:由多个专家分别生成人工摘要,构成标准摘要集,将系统生成的自动摘要与人工生成的标准摘要相对比,通过统计二者之间重叠的基本单元(*n* 元语法、词序列和词对)的数目,来评价摘要的质量.通过与标准人工摘要的对比,提高评价系统的稳定性和健壮性.该方法现已成为自动评价技术的通用标准之一.本文采用 ROUGE 中 ROUGE-1,ROUGE-2 和 ROUGE-L 来对生成的摘要进行评价.

- 对比系统

为了评估我们提出的模型(global self-matching mechanism,简称为 GSM)在自动摘要任务中的表现,需将 GSM 与下列自动摘要方法进行比较,比较时直接使用以下方法在原始论文中给出的在 LCSTS 数据集上的实验结果(其中,RNN 与 RNN-context 为 LCSTS 数据集论文中提供的基线模型).

(1) RNN[5]:使用 RNN 作为编码器,它的最后隐藏状态作为解码器的输入,在解码期间不使用本地上下文;
(2) RNN-context[5]:使用 RNN 作为编码器,在解码过程中使用上下文,编码器的所有隐藏状态的组合作为解码器的输入;
(3) CopyNet[16]:基于注意力机制的序列到序列模型,添加了拷贝机制.编码端为双向 RNN 结构,解码端为包含生成模式和拷贝模式的混合模型,生成模式从预设词表中选词,拷贝模式从输入序列中选词;
(4) SRB[19]:引入了基于语义相关性的神经模型来鼓励文本和摘要之间的高语义相似性.该模型由 3 部分组成——编码器、解码器和相似性函数:编码器将原文本压缩成语义向量;解码器生成摘要并产生所生成的摘要的语义向量;最后,相似度函数评估原文本的语义向量与生成的摘要之间的相关性,表示之间的相似性得分在训练期间使其最大化;
(5) DRGD[14]:在基于注意力机制的序列到序列模型的基础上,增加了潜在结构向量来学习目标摘要中隐含的潜在结构信息,以提高摘要质量;
(6) R-NET[24]:基于门控自匹配网络的阅读理解模型.由于原论文中的模型主要针对阅读理解任务,我们将其输出端改为基于注意力机制的 LSTM 解码器,以适应自动文摘任务;
(7) CGU[16]:引入了一种全局编码框架,它根据源端输入文档的全局信息控制从编码器到解码器的信息流,其由卷积门控单元组成,用于执行全局编码以改进源端信息的表示.

- 超参设置

我们使用 PyTorch(https://pytorch.org/)深度学习框架编写代码,并在 NVIDIA 1080Ti GPU 上进行实验.由于按词分割文本导致词表过大,在生成摘要时出现了大量未登录词,因此,本文使用按字分割文本,使用 unk 来表示所有字表的未登录字.我们使用默认设定参数的 Adam 优化器[30]:$lr$=0.001,$betas$=(0.9,0.999),$eps$=$1\times10^{-8}$.其他具体参数设置见表 3.其中,学习速率在第 8 轮迭代时开始减半.所有超参数都使用开发集进行调整,实验结果在测试集上报告.

**Table 3** Hyperparameters setting

**表 3** 实验参数设置表

| | |
|---|---|
| 字嵌入维度 | 300 维 |
| 字表大小 | 5 000 |
| LSTM 隐藏单元维度 | 600 维 |
| 编码器中 LSTM 的层数 | 2 层 |
| 解码器中 LSTM 的层数 | 2 层 |
| 批处理大小 | 64 |

### 3.2 实验结果及分析

本节首先将 GSM 系统和一些基准系统进行比较,然后分析不同层次的全局信息对生成式自动文摘模型的影响.最后,分析了本文模型中不同组件对模型的贡献情况.

(1) 与对比系统比较

为了评估提出的模型在自动摘要任务中的表现,我们将 GSM 与当前主流的自动摘要方法进行比较.实验结果见表 4,其中,GSM 为在 seq2seq 模型基础上添加了全局自匹配层和全局门控单元的模型.

**Table 4** Automatic evaluation result in LCSTS

**表 4** LCSTS 数据集上的实验结果

| Model | 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 |
| DRGD | 37.0 | 24.2 | 34.2 |
| R-NET | 37.8 | 25.3 | 35.0 |
| CGU | 39.4 | 26.9 | 36.5 |
| **GSM** | **41.1** | **28.5** | **38.3** |

首先,对比表 4 的实验结果可以看出:本文提出的 GSM 模型超过了所有对比的基准系统,达到了最好的效果,说明本文提出的基于全局自匹配机制与全局门控单元的 GSM 模型是有效的.据文献报道,在 LCSTS 语料上达到目前已知的最好效果的模型为 CGU,其根据源端上下文的全局信息控制从编码器到解码器的信息流,利用门控单元对源端的输入信息进行筛选.不过,CGU 模型仅根据全局信息对源端的输入信息进行筛选,没有将源端信息和全局信息进行有效地融合.相似的,R-NET 模型也通过一个门控单元对文章内容进行筛选.而本文提出的 GSM 模型在获取全局信息后,将源端每个词的隐层表示和其对应的全局信息进行了有效地融合,用于解码器生成摘要.根据表 4 实验结果,与 R-NET 和 CGU 模型相比,GSM 模型的性能有显著提高.特别地,与 CGU 模型相比,GSM 模型在 Rouge-1,Rouge-2 和 Rouge-L 上分别提高了 1.7,1.6 和 1.9 个百分点,说明将源端信息和全局信息进行有效融合对于文摘任务是必要的.

(2) 不同层次全局信息的差异分析

为了分析不同层次的全局信息对自动文摘模型的影响,我们对使用不同层次全局信息的模型进行比较.实验结果见表 5,其中,Uni-seq2seq 和 seq2seq 为我们复现的基于注意力机制的序列到序列模型,其具体实现方法为本文第 2 节描述的基于编码器-解码器架构的序列到序列学习模型(无全局自匹配层和全局门控单元).两者的不同之处在于,Uni-seq2seq 模型使用单向 LSTM 编码器,seq2seq 模型使用双向 LSTM 编码器.

read-again 模型为我们根据 Zeng 等人[15]论文复现的模型(LSTM 版),该模型将第 1 次编码时第一以及最后一个时间步对应隐层状态作为固定不变的全局信息,并将其作为第 2 次编码过程中每一次输入的附加输入来指导第 2 次编码.

**Table 5** Performances of various models with different levels of global information

**表 5** 具有不同层次全局信息的各种模型的性能

| Model | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Uni-seq2seq | 33.8 | 20.9 | 30.7 |
| seq2seq | 35.1 | 22.7 | 32.5 |
| read-again | 35.9 | 23.0 | 33.3 |
| **seq2seq+Match** | **39.7** | **26.9** | **36.6** |

从表 5 中的实验结果可以看出:

1) 具有完整上下文信息的模型优于仅包含部分上下文信息的模型.例如,使用双向 LSTM 编码器的 seq2seq 模型将编码器在每个时间步上获得的前后向信息进行拼接,以获得完整的上下文信息,在 ROUGE-1,ROUGE-2 和 ROUGE-L 评价指标上均优于使用单向 LSTM 编码器的 Uni-seq2seq 模型,其编码器每个时间步对应的隐层状态仅包含单向的、不完整的上下文信息;
2) 全局信息能够指导编码器进行更好的编码.不同于 seq2seq 模型仅仅将编码器在每个时间步上获得的前后向信息进行拼接以获得完整的上下文信息,seq2seq+Context 模型在第 2 次编码时,将第 1 次编码获得的全局信息作为第 2 次编码过程中每一次输入的附加输入,对第 2 次编码进行指导;同时,将全局信息编码入每一个时间步的隐层表示中.而根据表 5 中给出的实验结果可以发现,seq2seq+Context 模型在 Rouge-1,Rouge-2 和 Rouge-L 评价指标上均优于 seq2seq 模型,说明全局信息能够有效指导编码器进行更好地编码;
3) 动态的全局信息优于固定的全局信息.由于每个词关注的全局信息往往有所不同,因此为每一个词提供相同的全局信息存在一定的局限性.相比于 seq2seq+Context 模型,其使用第 1 次编码时最后一个时间步对应隐层状态作为全局信息,对于每一个词都是固定的,seq2seq+Match 模型使用本文提出的全局自匹配机制动态地从整个输入文本中收集与该单词相关的信息,并将单词表示和相关信息融合到最终隐层表示中.实验结果表明,seq2seq+Match 模型优于 seq2seq+Context 模型,在 Rouge-1,Rouge-2 和 Rouge-L 上分别提高了 3.8,3.9 和 3.3 个百分点.

(3) 不同组件的贡献分析

为了分析模型中不同组件对模型的贡献程度,我们在基础的 seq2seq 模型上分别加入全局自匹配机制和全局门控单元,并进行比较.实验结果见表 6,其中,seq2seq 模型与表 5 中相同,seq2seq+Gate 为在 seq2seq 模型基础上添加了本文提出的全局门控单元的模型.由于没有自匹配层,无法获得包含全局信息的隐层表示,其全局门控单元中的 $Q$,$K$ 和 $V$ 同为原始隐层表示 $h$.seq2seq+Match 为在 seq2seq 模型基础上添加了本文提出的自匹配层的模型,无全局门控单元.

**Table 6** Performance of various models with different components

**表 6** 具有不同组件的各种模型的性能

| Model | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| seq2seq | 35.1 | 22.7 | 32.5 |
| seq2seq+Gate | 38.3 | 25.7 | 35.2 |
| seq2seq+Match | 39.7 | 26.9 | 36.6 |
| **GSM** | **41.1** | **28.5** | **38.3** |

对比表 6 中的实验结果可以看出,模型中不同组件的贡献有所不同.

- 首先,在基础的基于注意力机制的序列到序列模型(seq2seq)上分别加入本文提出的全局门控单元(gate)和全局自匹配机制(match)后,在 ROUGE 评价指标上都有显著的提升,其中,seq2seq+Gate 模型在 Rouge-1,Rouge-2 和 Rouge-L 上分别提高了 3.2,3.0 和 2.7 个百分点,seq2seq+Match 模型在 Rouge-1,Rouge-2 和 Rouge-L 上分别提高了 4.6,4.2 和 4.1 个百分点,略优于 CGU 模型,说明本文提出的全局门控单元和全局自匹配机制是有效的;
- 其次,相比于 seq2seq+Gate 模型,seq2seq+Match 模型在 Rouge-1,Rouge-2 和 Rouge-L 上分别提高了 1.4,

1.2 和 1.4 个百分点,说明全局自匹配机制的作用大于全局门控单元,体现了有效地融合源端全局信息对摘要任务的重要性.虽然全局门控单元能根据全局信息对源端的信息进行筛选,过滤冗余信息,但是无法弥补没有有效融合源端信息造成的信息缺失;而全局自匹配机制虽然能将源端的全局信息有效地融合到源端每个词的隐层表示中,但是容易造成信息冗余;

- 最后,本文提出的 GSM 模型(即 seq2seq+Match+Gate)性能相比于单独的全局门控单元和全局自匹配机制都有显著提高,达到了最好的效果,说明其能较好地结合全局门控单元和全局自匹配机制两者的优点,并且有效地弥补它们各自存在的缺点,既可以获得完整的全局信息,同时又避免了信息冗余.

### 3.3 实验结果分析

表 7 中列出了 3 个示例,其包含了分别由 seq2seq,CGU,seq2seq+Match,seq2seq+Gate 和 GSM 这 5 个模型生成的摘要.

**Table 7** Some examples of the summary

**表 7** 部分生成摘要示例

| | |
|---|---|
| 文本 | 雅虎发布 2014 年第 4 季度财报,并推出了免税方式剥离其持有的阿里巴巴集团 15%股权的计划,打算将这一价值约 400 亿美元的宝贵投资分配给股东.截止发稿前,雅虎股价上涨了大约 7%,至 51.45 美元 |
| 参考摘要 | 雅虎宣布剥离阿里巴巴股份 |
| CGU | 阿里集团 400 亿美元收购阿里巴巴集团股权计划 |
| seq2seq | 雅虎发布阿里 15%股权的计划,雅虎股价上涨 7% |
| +Gate | 雅虎宣布 400 亿美元收购阿里巴巴 15%股权 |
| +Match | 雅虎宣布剥离阿里巴巴集团 15%股权 |
| **GSM** | **雅虎宣布剥离阿里巴巴股份** |
| 文本 | 国务院法制办公布《公共场所控制吸烟条例(送审稿)》:禁止所有烟草广告、促销和赞助;没有设置室外吸烟点的视为全面禁止吸烟;违反《条例》规定,电影电视剧播放吸烟镜头最高罚 3 万;禁止通过自动售货机等任何方式向未成年人售烟 |
| 参考摘要 | 我国拟全面禁止烟草广告影视剧播吸烟,最高罚 3 万 |
| CGU | 我国拟规定影视剧播放吸烟镜头最高罚 3 万 |
| seq2seq | 我国拟禁止向未成年人吸烟镜头,最高罚 3 万 |
| +Gate | 我国拟全面禁止烟草广告 |
| +Match | 我国拟全面禁止烟草广告,电视剧播放吸烟镜头最高罚 3 万 |
| **GSM** | **我国拟全面禁止烟草广告,影视剧播吸烟镜头最高罚 3 万** |
| 文本 | 近年来,逢雨必涝、逢涝必瘫,几成我国城市通病.上周,中国青年报对全国 31 个省(区、市)5 375 人进行的调查显示,91.6%的人关注所在城市的排水问题;84.7%的受访者赞同,城市现代化更表现在地面之下,应加大地下民生工程建设投入 |
| 参考摘要 | 84.7%受访者期待国家加大地下民生工程投入 |
| CGU | 84.7%受访者认为城市现代化应加大民生工程投入 |
| seq2seq | 84.7%受访者赞同“逢雨必涝” |
| +Gate | 调查显示,91.6%受访者关注目前城市排水问题 |
| +Match | 调查显示,84.7%受访者赞同城市排水问题应加大地下民生工程建设 |
| **GSM** | **84.7%受访者认为,城市现代化应加大地下民生工程** |

通过对比分析可以观察到:在 3 个例子中,加入本文提出的全局自匹配机制的 seq2seq+Match 模型和 GSM 模型都较为全面地抓住了文章的主旨,而 seq2seq,CGU 和 seq2seq+Gate 模型往往只能抓住文摘主旨的一部分.相比于 seq2seq+Gate 模型,虽然 seq2seq+Match 能较为全面地抓住文章的主旨,但是往往存在冗余信息.在第 1 个例子中,GSM 模型生成的摘要与参考摘要完全一致,seq2seq+Match 模型也涵盖了文章主旨;而 Seq2Seq 模型虽然提及了“阿里 15%股权”,但是没有说明“剥离”这个动作;CGU 模型虽然提及了“阿里巴巴集团股权”,但是把“雅虎剥离”错误理解为“阿里集团收购”,导致生成错误摘要;类似的,seq2seq+Gate 模型把“雅虎剥离”错误理解为“雅虎宣布 400 亿美元收购”.在第 2 个例子中,seq2seq 和 CGU 模型都只考虑到了“吸烟镜头”,忽视了“烟草广告”,seq2seq+Gate 模型只考虑到了“烟草广告”,忽视了“吸烟镜头”,都存在信息的缺失;而 seq2seq+Match 和 GSM 模型生成的摘要则同时涵盖了“烟草广告”与“吸烟镜头”,与参考摘要也十分相似.在第 3 个例子中,seq2seq 模型没有抓住文摘主旨;Seq2Seq+Gate 模型关注点发生了偏移;CGU 模型基本包含文章主旨,却缺失了“地下”这个关键词;而 Seq2Seq+Match 和 GSM 模型仍然很好地抓住了文章主旨,也注意到了“地下民生工程”这个关键词.

不足的是,GSM 模型生成的摘要缺失了宾语“投入”,造成语法错误.另外,在 3 个例子中,GSM 模型生成的摘要相对于 Seq2Seq+Match 模型生成的摘要更简洁一些,去除了一些冗余信息,更接近于参考摘要.

## 4 结论与未来的工作

自动文摘是自然语言处理领域的重要研究方向之一,近 60 年持续性的研究,已经在部分自动文摘任务上取得了明显进展.本文对当前主流的基于编码器-解码器架构的序列以及序列学习模型进行了改进:在利用传统的编码器对输入文本进行编码之后,增加了全局自匹配层对编码后的输入文本进行自匹配的过程,能够动态地从整个输入文本中为文本中每一个词收集与该词相关的信息,并将该词及其匹配的全局信息编码到最终的表示中.同时,增加了全局门控单元对自匹配层获得的表示进行的步骤筛选,去除冗余信息,以便挖掘出原文档的核心内容.在 LCSTS 语料上的实验表明,与当前主流的生成式摘要方法相比,该方法在 ROUGE 评价指标上有显著提高.

因短文本摘要的原文字数少,且只有一个段落,因此本文使用的全局信息为整个全文(篇章)的信息,可以理解为利用了篇章的物理结构信息.由于长文本摘要的原文有多个段落,未来工作中,我们将从语言学角度出发,在考虑利用篇章的物理结构信息的基础上,还将考虑篇章的语义结构(如篇章修辞结构、话题结构等)信息来生成长文本摘要.由于当前主流的神经网络框架尚不能够有效地对长文档进行语义编码,针对这一问题,未来的工作将尝试通过分析篇章的衔接性和连贯性,从整体上分析出篇章结构及其构成单元之间的语义关系,并利用上下文理解篇章,来辅助篇章级长文本的摘要生成,弥补传统神经网络框架对长文档语义编码的不足.

**References**:


[1] Luong T, Pham H, Manning CD. Effective approaches to attention-based neural machine translation. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: The Association for Computational Linguistics, 2015. 1412−1421.

[2] Graves A, Mohamed A, Hinton G. Speech recognition with deep recurrent neural networks. In: Proc. of the 38th IEEE Int’l Conf. on Acoustics, Speech, and Signal Processing. Vancouver: IEEE, 2013. 6645−6649.

[3] Zhou Q, Yang N, Wei F, Zhou M. Selective encoding for abstractive sentence summarization. In: Proc. of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: The Association for Computational Linguistics, 2017. 1095−1104.

[4] Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Trans. on Signal Processing, 1997,45(11):2673−2681.

[5] Hu B, Chen Q, Zhu F. LCSTS: A large scale chinese short text summarization dataset. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: The Association for Computational Linguistics, 2015. 1967−1972.

[6] Chali Y, Hasan SA, Joty SR. A SVM-based ensemble approach to multi-document summarization. In: Proc. of the Advances in Artificial Intelligence, 22nd Canadian Conf. on Artificial Intelligence. Kelowna: Springer-Verlag, 2009. 199−202.

[7] Siddharthan A, Nenkova A, McKeown K. Syntactic simplification for improving content selection in multi-document summarization. In: Proc. of the 20th Int’l Conf. on Computational Linguistics. Geneva: The Association for Computational Linguistics, 2004. 896−1003.

[8] Mihalcea R, Tarau P. TextRank: Bringing order into text. In: Proc. of the 2004 Conf. on Empirical Methods in Natural Language Processing. Barcelona: The Association for Computational Linguistics, 2004. 404−411.

[9] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. In: Proc. of the 26th Int’l Conf. on Neural Information Processing Systems. Lake Tahoe: Neural Information Processing Systems, 2013. 3111−3119.

[10] Blitzer J, Weinberger KQ, Saul LK, Pereira F. Hierarchical distributed representations for statistical language modeling. In: Proc. of the 17th Int’l Conf. on Neural Information Processing Systems. Vancouver: Neural Information Processing Systems, 2004. 185−192.

[11] Rush AM, Chopra S, Weston J. A neural attention model for abstractive sentence summarization. In: Proc. of the 2015 Conf. on Empirical Methods in Natural Language Processing. Lisbon: The Association for Computational Linguistics, 2015. 379−389.

[12] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.

[13] Chopra S, Auli M, Rush AM. Abstractive sentence summarization with attentive recurrent neural networks. In: Proc. of the 2016 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego California: The Association for Computational Linguistics, 2016. 93−98.

[14] Li P, Lam W, Bing L, Wang Z. Deep recurrent generative decoder for abstractive text summarization. In: Proc. of the 2017 Conf. on Empirical Methods in Natural Language Processing. Copenhagen: The Association for Computational Linguistics, 2017. 2091−2100.

[15] Zeng W, Luo W, Fidler S, Urtasun R. Efficient summarization with read-again and copy mechanism. arXiv preprint arXiv:1611.03382, 2016.

[16] Gu J, Lu Z, Li H, Li VOK. Incorporating copying mechanism in sequence-to-sequence learning. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: The Association for Computational Linguistics, 2016. 1631−1640.

[17] See A, Liu PJ, Manning CD. Get to the point: summarization with pointer-generator networks. In: Proc. of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: The Association for Computational Linguistics, 2017. 1073−1083.

[18] Lin J, Sun X, Ma S, Su Q. Global encoding for abstractive summarization. In: Proc. of the 56th Annual Meeting of the Association for Computational Linguistics. Melbourne: The Association for Computational Linguistics, 2018. 163−169.

[19] Ma S, Sun X, Li W, Li S, Li W, Ren X. Query and output: generating words by querying distributed word representations for paraphrase generation. In: Proc. of the 2018 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. New Orleans: The Association for Computational Linguistics, 2018. 196−206.

[20] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997,9(8):1735−1780.

[21] Mikolov T, Karafiát M, Burget L, Cernocký J, Khudanpur S. Recurrent neural network based language model. In: Proc. of the Eleventh Annual Conf. of the Int'l Speech Communication Association. Makuhari: ISCA, 2010. 1045−1048.

[22] Rocktäschel T, Grefenstette E, Hermann KM, Kociský T, Blunsom P. Reasoning about entailment with neural attention. arXiv preprint arXiv:1509.06664, 2015.

[23] Wang S, Jiang J. Machine comprehension using match-LSTM and answer pointer. arXiv preprint arXiv:1608.07905, 2016.

[24] Wang W, Yang N, Wei F, Chang B, Zhou M. Gated self-matching networks for reading comprehension and question answering. In: Proc. of the 55th Annual Meeting of the Association for Computational Linguistics. Vancouver: The Association for Computational Linguistics, 2017. 189−198.

[25] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I. Attention is all you need. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Neural Information Processing Systems, 2017. 5998−6008.

[26] Gulcehre C, Ahn S, Nallapati R, Zhou B, Bengio Y. Pointing the unknown words. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: The Association for Computational Linguistics, 2016. 140−149.

[27] Bengio S, Vinyals O, Jaitly N, Shazeer N. Scheduled sampling for sequence prediction with recurrent neural networks. In: Proc. of the 28th Int'l Conf. on Neural Information Processing Systems. Montreal: Neural Information Processing Systems, 2015. 1171−1179.

[28] Koehn P. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In: Proc. of the Conf. of the Association for Machine Translation in the Americas 2004. Washington: Springer, 2004. 115−124.

[29] Lin CY, Hovy E. Automatic evaluation of summaries using *N*-gram co-occurrence statistics. In: Proc. of the Conf. of the North American Chapter of the Association for Computational Linguistics on Human Language Technology. Edmonton: The Association for Computational Linguistics, 2003. 71−78.

[30] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.

**吴仁守**(1995－),男,浙江温州人,硕士生,CCF 学生会员,主要研究领域为自然语言处理,信息抽取.

**王中卿**(1987－),男,博士,CCF 专业会员,主要研究领域为自然语言处理.

**王红玲**(1975－),女,博士,副教授,CCF 专业会员,主要研究领域为自然语言处理,信息抽取.

**周国栋**(1967－),男,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为自然语言处理,信息抽取.