



北京大学学报(自然科学版)

Acta Scientiarum Naturalium Universitatis Pekinensis

ISSN 0479-8023,CN 11-2442/N

《北京大学学报(自然科学版)》网络首发论文

题目: 基于语义对齐的生成式自动摘要研究
作者: 吴世鑫, 黄德根, 李玖一
DOI: 10.13209/j.0479-8023.2020.084
收稿日期: 2020-05-15
网络首发日期: 2020-10-14
引用格式: 吴世鑫, 黄德根, 李玖一. 基于语义对齐的生成式自动摘要研究. 北京大学学报(自然科学版). <https://doi.org/10.13209/j.0479-8023.2020.084>



网络首发: 在编辑部工作流程中, 稿件从录用到出版要经历录用定稿、排版定稿、整期汇编定稿等阶段。录用定稿指内容已经确定, 且通过同行评议、主编终审同意刊用的稿件。排版定稿指录用定稿按照期刊特定版式(包括网络呈现版式)排版后的稿件, 可暂不确定出版年、卷、期和页码。整期汇编定稿指出版年、卷、期、页码均已确定的印刷或数字出版的整期汇编稿件。录用定稿网络首发稿件内容必须符合《出版管理条例》和《期刊出版管理规定》的有关规定; 学术研究成果具有创新性、科学性和先进性, 符合编辑部对刊文的录用要求, 不存在学术不端行为及其他侵权行为; 稿件内容应基本符合国家有关书刊编辑、出版的技术标准, 正确使用和统一规范语言文字、符号、数字、外文字母、法定计量单位及地图标注等。为确保录用定稿网络首发的严肃性, 录用定稿一经发布, 不得修改论文题目、作者、机构名称和学术内容, 只可基于编辑规范进行少量文字的修改。

出版确认: 纸质期刊编辑部通过与《中国学术期刊(光盘版)》电子杂志社有限公司签约, 在《中国学术期刊(网络版)》出版传播平台上创办与纸质期刊内容一致的网络版, 以单篇或整期出版形式, 在印刷出版之前刊发论文的录用定稿、排版定稿、整期汇编定稿。因为《中国学术期刊(网络版)》是国家新闻出版广电总局批准的网络连续型出版物(ISSN 2096-4188, CN 11-6037/Z), 所以签约期刊的网络版上网络首发论文视为正式出版。

北京大学学报(自然科学版)

Acta Scientiarum Naturalium Universitatis Pekinensis

doi: 10.13209/j.0479-8023.2020.084

基于语义对齐的生成式自动摘要研究

吴世鑫 黄德根[†] 李玖一

大连理工大学计算机学院, 大连 116023; [†] 通信作者, E-mail: huangdg@dlut.edu.cn

摘要 针对当前生成式自动摘要模型在解码时对摘要整体语义信息利用不充分的问题, 提出一种基于语义对齐的神经网络自动摘要方法。该方法以带注意力、Pointer 机制和 Coverage 机制的 Sequence-to-Sequence 模型为基础, 在编码器和解码器间加入语义对齐网络, 实现文本到摘要的语义信息对齐; 将获得的摘要整体语义信息与解码器的词汇预测上下文向量进行拼接, 使解码器在预测当前词汇时不仅利用已预测词汇序列的部分语义, 而且考虑拟预测摘要的整体语义。在中文新闻语料 LCSTS 上的实验表明, 该模型能够有效提高自动摘要的质量。
关键词 生成式文本摘要; Sequence-to-Sequence 模型; 语义对齐网络

Abstractive Summarization Based on Semantic Alignment Network

WU Shixin, HUANG Degen[†], LI Jiuyi

Dalian University of Technology, Dalian 116023; [†] Corresponding author, E-mail: huangdg@dlut.edu.cn

Abstract Aiming at the problem of insufficient utilization of the overall semantic information of abstracts in decoding by the currently abstractive summarization model, this paper proposes a neural network automatic abstract model based on semantic alignment. This model is based on the Sequence-to-Sequence model with attention, Pointer mechanism and Coverage mechanism. A semantic alignment network is added between the encoder and the decoder to achieve the semantic information alignment of the text to the abstract. The achieved semantic information is concatenated with the context vector at the time of decoding, so that when the decoder predicts the vocabulary, it not only uses the partial semantics before the time of decoding, but also considers the overall semantics of the digest sequence. Experiments on the Chinese news corpus LCSTS show that the proposed model can effectively improve the quality of automatic summarization.

Key words abstractive summarization; sequence-to-sequence model; semantic alignment network

文本自动摘要是指计算机自动生成准确全面反映某一文本中心内容的简洁连贯短文的过程^[1]。文本摘要自动生成技术应用广泛, 尤其是在提高用户获取信息效率和实现文本压缩存储方面的作用越来越突出。文本自动摘要的分类方式有很多, 按照摘要生成方法可以分为抽取式摘要和生成式摘要。抽取式摘要是指从文本中抽取现有的若干句子组合成为文本的摘要, 而生成式摘要则是综合分析原文信息后通过算法自动生成新句子作为文本的摘要。

由于生成式方法更贴近人工生成摘要的过程, 且在语法准确度和语义连贯性上较抽取式方法更有优势, 因此受到越来越多的重视。目前生成式方法通常以序列到序列(Sequence-to-Sequence)深度神经网络模型^[2]为基础, 该模型的作用是在编码器-解码器的框架下, 将源序列转化为目标序列。对于文本摘要自动生成任务而言, 就是将文本词汇序列转化为摘要词汇序列。生成式文本自动摘要技术近年来得到了长足发展, 其中很多研究尝试对编码端进行改进, 不断挖掘并细化对文本有效信息的利用程度。Lin 等^[3]提出一种基于卷

积神经网络(Convolutional Neural Network, CNN)和自注意力机制的全局编码门,提高对文本词汇特征和内部联系的挖掘;Nallapati 等^[4]以带注意力机制的 Sequence-to-Sequence 模型为基础,利用语言特征构造词向量,并将其与普通词向量相结合,丰富了编码器词嵌入时包含的信息;Chopra 等^[5]利用卷积工具在原词汇编码上增加其位置和上下文信息,提出条件循环神经网络模型;Wang 等^[6]在编码器文本表示部分构建检索、重排序和 BiSET 三个模型,通过检索和重排序构建文本表示模板,再利用 BiSET 结合模板与文本构建编码器端的文本表示,从而提高文本摘要质量。此外,还有部分研究从解码端作为切入点,通过增强解码器词汇预测的准确性提高摘要生成质量。其中,Sebastian Gehrmann 等^[7]针对解码器在文本内容选择上的局限性,提出一种短语粒度的内容选择器作为自底向上的注意力,使解码器更多关注文本中可作为摘要的短语;Song 等^[8]在词向量基础上扩展了词汇结构特征信息,并将句子语法结构融入解码器的注意力机制中,使结构化的关键词和其在文本中的语法关系保留到摘要词汇预测序列中,帮助再现事实细节;See 等^[9]为解决解码器无法生成词典外词汇(Out of Vocabulary, OOV)以及预测词汇重复的问题,分别提出了指针机制和覆盖机制,提高了生成摘要的质量;Tan 等^[10]提出了一种基于图的注意力机制,提高解码器对文本中显性信息的挖掘程度,并在解码阶段利用一种分层集束搜索算法生成多句摘要;Cao 等^[11]利用相似句子有相似摘要的假设,在解码器中引入由检索、重排序和重写三个部分组成的软模板,首先通过检索得到相似句,然后利用重排序进行排序,最后重写利用排序结果进行摘要词汇预测。以上对解码器进行改进的方法均提高了摘要生成质量,但解码器在解码时往往仅对已预测的一个或多个词汇语义进行片段性利用,并未有效利用预测摘要的整体语义信息。

针对上述问题,本文提出一种基于语义对齐的神经网络自动摘要模型,以带注意力、Coverage 机制和 Pointer 机制的 Sequence-to-Sequence 模型为基础,在编码端和解码端间加入语义对齐网络。通过该语义对齐网络挖掘预测摘要的整体语义信息,并在解码端进行词汇预测时将该整体语义信息与上下文向量进行拼接,从而丰富上下文向量表示,并提高词汇预测时的准确性。实验证明,该方法能够提高自动摘要的生成质量。

1 相关工作

1.1 指针网络(pointer network)

指针网络^[9]是在注意力机制基础上的改进,提出了 Pointer 机制,其注意力分布计算具体过程如下:

$$e_{t,i} = V^T \tanh(W^e [s_{t-1}; h_i^e] + b^e), \quad (1)$$

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^m \exp(e_{t,j})}, \quad (2)$$

$$C_t = \sum_{i=1}^m \alpha_{t,i} h_i^e, \quad (3)$$

其中, h_i^e 表示编码端输入序列中第 i 个词的隐状态; s_{t-1} 表示解码端在 t 时刻输入的解码隐状态; m 为输入序列的长度; $\alpha_{t,i}$ 表示注意力分布,即解码器在 t 时刻预测摘要词汇时对文本第 i 个词汇的关注情况; C_t 表示 decoder 在 t 时刻的上下文向量; V, W^e, b^e 均为可训练的参数。

接着,上下文向量 C_t 与当前时刻解码状态 s_t 结合生成词汇分布概率:

$$P_{vocab}(w) = \text{softmax}(W^y [C_t; s_t] + b^y), \quad (4)$$

其中, w 表示要预测的词汇; P_{vocab} 是词典中词汇的概率分布; W^y, b^y 均为可训练参数。

以上是基于注意力机制的词预测的一般过程, Pointer 机制的目的是允许解码器可以通过从词典中生成和复制输入词汇两种方式进行词预测,具体过程如下:

首先计算在给定上下文向量 C_t 、 t 时刻解码器隐状态 s_t 和解码器 $t-1$ 时刻预测序列 y_{t-1} 条件下进行词

生成的概率:

$$p_{gen} = \sigma(W_h^T C_t + W_s^T s_t + W_y^T y_{t-1} + b_{ptr}), \quad (5)$$

其中, $W_h^T, W_s^T, W_y^T, b_{ptr}$ 均为可训练参数。

接着, p_{gen} 作为开关控制通过生成或复制的方式预测词, 利用公式(6)计算在 Pointer 机制下的词预测概率:

$$p(w) = p_{gen} p_{vocab}(w) + (1 - p_{gen}) \sum_{i: w_i = w} \alpha_{t,i}. \quad (6)$$

1.2 覆盖机制(coverage mechanism)

在传统的基于注意力机制的 Sequence-to-Sequence 模型中, 预测词汇出现重复也是一个普遍存在的问题。究其原因, 与解码器解码时注意力重复集中在同一源词汇有很大关系, Coverage 机制^[9]正是用于解决注意力重复问题。

Coverage 机制的实现是通过在模型中构建一个覆盖向量 c_i^t , 该覆盖向量的构建方法如公式(7)所示:

$$c_i^t = \sum_{k=0}^{t-1} \alpha_{k,i}. \quad (7)$$

从构建方法可以看出, 覆盖向量表示解码器 t 时刻之前对文本第 i 个词汇注意力分布的总和。覆盖向量生成后原来的注意力计算公式(1)即改为:

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + W_c c_i^t + b_{att}), \quad (8)$$

其中, $v^T, W_c, W_s, W_c, b_{att}$ 为可训练参数。新的注意力计算公式可以确保计算当前时刻注意力时将之前所有的注意力分布情况考虑在内。

由此也需要在总的损失函数中加入覆盖损失, 具体计算方法如公式(9)所示:

$$covloss_t = \sum_i \min(\alpha_{t,i}, c_i^t). \quad (9)$$

2 基于语义对齐的生成式自动摘要模型

以带注意力、Pointer 和 Coverage 机制的 Sequence-to-Sequence 模型为基础, 通过添加语义对齐网络, 构造基于语义对齐的神经网络自动摘要模型, 模型的输入为文本词汇序列 x , 输出为摘要词汇序列 y , 完整结构如图 1 所示。

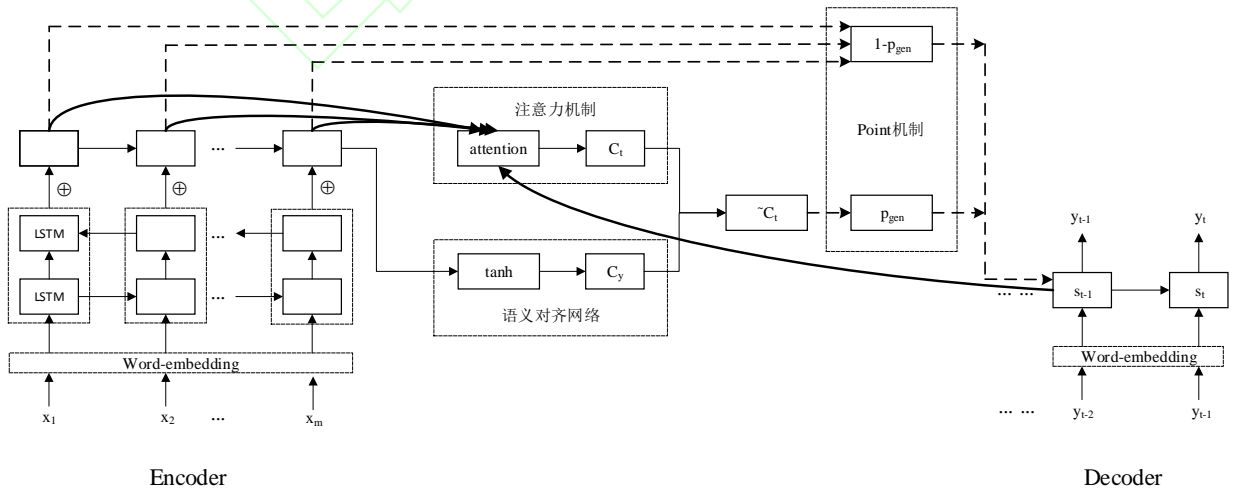


图 1 基于语义对齐的神经网络自动摘要模型

Fig. 1 Automatic summarization model of neural network based on semantic alignment

2.1 编码端

编码端采用两层长短期记忆网络(Long Short-Term Memory, LSTM)网络结构, 第一层为双向 LSTM 网络, 第二层为单向 LSTM 网络。构建过程如下列公式所示:

$$\vec{h}_i = LSTM(\vec{h}_{i-1}, x_i), \quad (10)$$

$$\overleftarrow{h}_i = LSTM(x_i, \overleftarrow{h}_{i+1}), \quad (11)$$

$$h_i^e = \overrightarrow{h}_i^e = LSTM(h_{i-1}^e, [\vec{h}_i; \overleftarrow{h}_i]), \quad (12)$$

其中, \vec{h}_i 和 \overleftarrow{h}_i 分别表示输入序列在第一层双向 LSTM 网络得到的前向和反向隐状态, h_i^e 表示第二层 LSTM 网络的隐状态。

2.2 解码端

解码端采用单层单向 LSTM 网络结构, 构建过程如下列公式所示:

$$s_t = LSTM(s_{t-1}, y_{t-1}), \quad (13)$$

其中, s_t 表示 t 时刻的解码隐状态。

1) 注意力机制

$$e_{t,i} = V^T \tanh(W^e [s_{t-1}; h_i^e] + b^e), \quad (14)$$

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{j=1}^m \exp(e_{t,j})}, \quad (15)$$

其中, m 为输入序列的长度; V, W^e, b^e 均为可训练的参数。

2) Coverage 机制

$$c_i^t = \sum_{k=0}^{t-1} \alpha_{k,i}, \quad (16)$$

$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + W_c c_i^t + b_{atm}), \quad (17)$$

$$a_{t,i} = \frac{\exp(e_i^t)}{\sum_{j=1}^m \exp(e_j^t)}, \quad (18)$$

$$C_t = \sum_{i=1}^m a_{t,i} h_i^e, \quad (19)$$

其中, C_t 表示解码器在 t 时刻进行词汇预测的上下文向量; $v^T, W_c, W_s, W_e, b_{atm}$ 均为可训练参数。

3) 语义对齐网络语义对齐网络(Semantic Alignment Network, SAN)通过文本和摘要之间的语义对齐, 丰富解码器在预测词汇时上下文向量的语义信息, 构造过程如下:

$$C_y = \tanh(W_a h_m^e + b_a), \quad (20)$$

$$C_t = [C_t; C_y], \quad (21)$$

其中, h_m^e 为 encoder 第二层 LSTM 网络的末尾隐状态, m 为输入序列的长度; W_a, b_a 为可训练参数。

4)Pointer 机制

$$P_{vocab}(y) = \text{soft max} \left(W^y \left[C_t; s_t \right] + b^y \right), \quad (22)$$

$$p_{gen} = \sigma \left(W_h^T C_t + W_s^T s_t + W_y^T y_{t-1} + b_{ptr} \right), \quad (23)$$

$$p(y) = p_{gen} p_{vocab}(y) + (1 - p_{gen}) \sum_{i: w_i = w} a_{t,i}, \quad (24)$$

其中, $W^y, b^y, W_h^T, W_s^T, W_y^T, b_{ptr}$ 均为可训练参数。

2.3 损失函数

$$L = -\frac{1}{n} \sum_{t=1}^{n+1} \log P(y_t^* | y_1^*, \dots, y_{t-1}^*, x; \theta) + \sum_{t=1}^{n+1} \text{cov loss}_t, \quad (25)$$

其中, L 表示损失函数; y_t^* 表示预测词汇, n 为摘要序列长度; x 表示输入序列; θ 代表整个模型中的可训练参数。

3 实验结果与分析

3.1 语料

实验使用 LCSTS^[12](A Large Scale Chinese Short Text Summarization Dataset)语料, 其内容来自新浪微博, 共包括 3 个部分: 第 1 部分共包含 2,400,591 个文本-摘要序列; 第 2 部分包含 10,666 个带人工打分标签的文本-摘要序列; 第 3 部分包含 1,106 个人工交叉打分一致的文本-摘要序列。本文选取第 1 部分为训练集, 第 2 部分为验证集, 第 3 部分为测试集。

3.2 实验结果

实验从基于字和基于词两个粒度展开, 模型训练的相关参数设定如下: 输入和输出词(字)维度均为 128; LSTM 网络隐藏层维度 256; 学习率 0.1; batchsize 批次大小设定为 100; beamsize 集束搜索宽度设定为 4; 词典大小为 50,000; 字典大小为 10723; epoch 为 30; 结果评价工具采用 rouge^[13]。相关实验结果如表 1、表 2 所示。

表 1 基于词粒度的实验结果对比

模型	Rouge_1	Rouge_2	Rouge_L
RNN(baseline)	17.7	8.5	15.8
RNN+Attention+Pointer+Coverage(RAPC)	35.2	18.0	27.5
RAPC+SAN	35.3	18.0	26.2

表 1 显示, 在词粒度上, 基于注意力机制、Pointer 机制和 Coverage 机制的 RAPC 模型总体上比 baseline 结果要好, 在 RAPC 上加入语义对齐网络后 rouge_1 提高 0.1 个百分点, rouge_2 持平, 而 rouge_L 则稍降低了 1.3 个百分点。

表 2 基于字粒度的实验结果对比

模型	Rouge_1	Rouge_2	Rouge_L
RNN (baseline)	21.5	8.9	18.6
RAPC	33.0	16.1	24.1
RAPC+SAN	34.7	17.8	29.5

表 2 显示, 在字粒度上, RAPC 在 3 个评价指标上表现均比 baseline 好, 在 RAPC 上加入语义对齐网络后 rouge_1 提高 1.7 个百分点, rouge_2 提高 1.7 个百分点, 而 rouge_L 则提高 5.4 个百分点。

为检验语义对齐网络对 UNK(未登录词标识)和词汇重复问题的影响程度,选取 RAPC 和 RAPC+SAN 两个模型,对测试集 1106 个文本所生成摘要中 UNK 和出现重复词汇句子的数量进行统计,结果如表 3 所示。

表 3 加入语义对齐网络前后出现 UNK 和重复词汇情况

Table 3 UNK and repeated words before and after joining the semantic alignment network

模型	UNK/个	出现重复词汇句子/个
RAPC(词粒度)	555	43
RAPC+SAN(词粒度)	526	40
RAPC(字粒度)	0	4
RAPC+SAN(字粒度)	0	3

表 4 人工评价结果

Table 4 Results of human evaluation

模型	信息吻合度	简洁性	可读性
RAPC(词粒度)	3.21	3.57	3.61
RAPC+SAN(词粒度)	3.25	3.58	3.69
RAPC(字粒度)	3.23	3.74	3.72
RAPC+SAN(字粒度)	3.57	3.89	3.92

另外,为检验语义对齐网络对摘要生成质量的影响,对 RAPC 和 RAPC+SAN 两个模型生成的摘要结果进行人工评价。具体做法为:首先,寻找 5 名评价人员(3 位研究生学历,2 位本科学历);其次,分别从两个模型的生成结果中随机选取 200 条摘要内容进行人工评价,评价内容包括生成摘要与参考摘要的信息吻合度、生成摘要的语言简洁性和可读性,从 1 到 5 进行打分,分数越高代表相应的性能越好;最后,对打分结果取均值。人工评价结果如表 4 所示。

最后,通过两个例句对摘要生成结果进行具体对比,如表 5 所示。

表 5 摘要生成结果对比

Table 5 Summary generation result comparison

原文	模型	结果
(例句 1)市场的阴霾开始扩散到投行从业人员的头上,那些怀揣着投行梦的年轻人,此时不得不重新检视自己当初的职业选择。裁员规模加大,津贴减少,女性需要在家庭和事业中取舍,竞争压力剧增。投行在年轻人心中一直像个梦般美好,而今,它越来越现实。	Reference	市场阴霾扩散投行梦难圆
	RAPC(词粒度)	投行的“<unk>”时代到来了吗???
	RAPC+SAN(词粒度)	投行梦 的“<unk>”:投行梦的<unk>之路?
	RAPC(字粒度)	投行梦的投行梦与投行梦的故事——投行
	RAPC+SAN(字粒度)	投行梦的年轻人不得不重新检视自己裁员
(例句 2)继去年底被曝出两位企业主负债跑路半年后,江阴再现老板跑路事件。5月中旬,江苏融泰石油科技股份有限公司实际控制人许才良出逃美国,使当地银行面临近 8 亿坏账风险。所涉贷款主要集中在工行、华夏银行、广发银行和江阴农商行。	Reference	江阴再现老板跑路牵涉近 8 亿银行资金
	RAPC(词粒度)	<unk>石油实际控制人许才良出逃美国面临 8 亿坏账风险风险!!!! 江阴
	RAPC+SAN(词粒度)	江阴再现 8 亿坏账风险银行面临近 8 亿坏账风险?!!!! 江阴
	RAPC(字粒度)	江阴再现老板跑路事件江苏融泰公司面临
	RAPC+SAN(字粒度)	江阴再现老板跑路事件 牵涉 8 亿 坏账贷款

3.3 实验分析

从表 1 和 2 可以看出,基于词粒度的实验结果提升不够明显,而基于字粒度的实验结果整体有了较大

改进,下面分别从字粒度和词粒度两个部分对实验结果展开分析。

词粒度实验方面,从人工评价结果(表 4)可以看出,加入语义对齐网络后,生成摘要在信息吻合度、简洁性和可读性上均有了一定提升,说明语义对齐网络能够改善生成摘要的质量。但是,从 UNK 和词汇重复统计结果(见表 3)注意到,加入语义对齐网络后 UNK 和词汇重复问题并未得到明显改善。词汇重复问题在模型中主要依靠 Coverage 机制进行规避,而 UNK 问题则跟词典大小和 Pointer 机制有很大关系,表 3 结果说明语义对齐网络不能很好解决以上两个问题,这在一定程度上影响了 rouge 值的提升,因此表 1 中加入语义对齐网络前后结果基本持平。从摘要生成结果(表 5)的示例也可以印证上述分析,加入语义对齐网络后,例句 1 中原来的“投行”变成了“投行梦”,例句 2 预测出“江阴再现”,与原文意思更加贴近,说明词汇生成的准确性有了提高。但是还存在大量词汇重复以及 UNK 的问题,如例句 1 中加入语义对齐网络后由“投行的“<unk>”时代到来了吗???”变成了“投行梦的“<unk>”: 投行梦的<unk>之路?”,词汇重复和 UNK 问题并未有效改善。

字粒度实验方面,从人工评价结果(表 4)可以看出,加入语义对齐网络后预测摘要在信息吻合度、简洁性和可读性上均有较大提升,尤其是生成摘要与参考摘要的信息吻合度提升 0.34。究其原因,基于字粒度的模型在生成摘要序列时使用的是字典,相较固定大小的词典能够显著提升词汇覆盖度,有效规避 UNK 问题。从 UNK 和词汇重复统计结果(见表 3)可以看出,基于字粒度的模型未出现 UNK,这种情况下解码器预测词汇时所使用的上下文语义信息在句子生成中的作用会更加突出。表 2 和 4 也表明,加入语义对齐网络后有效改善了生成摘要的整体质量。具体来看,在摘要生成结果(表 5)的示例中,加入语义对齐网络后,例句 1 中“投行梦的年轻人不得不重新检视自己裁员”相较原来的“投行梦的投行梦与投行梦的故事——投行”在可读性和简洁性上都有了很大提升,例句 2 预测出“牵涉 8 亿”,与参考摘要的信息吻合度更高,可见加入语义对齐网络后的确丰富了解码过程的整体语义,对于提升摘要效果具有较大帮助。

4 结论

针对生成式文本自动摘要模型中解码器对摘要整体语义利用不充分的问题,提出一种基于语义对齐的神经网络自动摘要模型,通过构造语义对齐网络提高了解码器在解码时上下文向量的语义信息表示能力,使解码器在预测当前词汇时不仅利用已预测词汇序列的部分语义,而且考虑了拟预测摘要的整体语义。实验结果表明,基于语义对齐的神经网络自动摘要模型能够提升生成摘要的准确性和连贯性。

参考文献

- [1] Gomaa W H, Fahmy A A. A survey of text similarity approaches. International Journal of Computer Applications: 2013, 68(13):13–18
- [2] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks // Advances in neural information processing systems. 2014: 3104–3112
- [3] Lin J, Xu S, Ma S, et al. Global Encoding for Abstractive Summarization // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2018: 163–169
- [4] Nallapati R, Zhai F, Zhou B. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents // Thirty-First AAAI Conference on Artificial Intelligence. 2017
- [5] Chopra S, Auli M, Rush A M. 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. 2016: 93–98
- [6] Wang K, Quan X, Wang R. BiSET: Bi-directional Selective Encoding with Template for Abstractive Summarization // Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 2153–2162
- [7] Gehrmann S, Deng Y, Rush A. Bottom-Up Abstractive Summarization // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 4098–4109
- [8] Song K, Zhao L, Liu F. Structure-Infused Copy Mechanisms for Abstractive Summarization // Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1717–1729

- [9] See A, Liu P J, Manning C D. Get to the point: Summarization with pointer-generator networks. Association for Computational Linguistics: 2017, 1099(17): 1073–1083
- [10] Tan J, Wan X, Xiao J. Abstractive document summarization with a graph-based attentional neural model // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017: 1171–1181
- [11] Cao Z, Li W, Li S, et al. Retrieve, rerank and rewrite: Soft template based neural summarization // Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018: 152–161
- [12] Hu B, Chen Q, Zhu F. LCSTS: A Large Scale Chinese Short Text Summarization Dataset // Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015: 1967–1972
- [13] Chopra S, Auli M, Rush A M. 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. 2016: 93–98

