

基于 NN-Attention 的中文短文本摘要^①



何正方, 梁宇

(云南大学 软件学院, 昆明 650500)

通讯作者: 何正方, E-mail: hfrommane@qq.com

摘要: 在以往的 Attention 模型中, 只采用了 Bidirectional-RNN, BRNN 对上下文信息是有效的, 但是无法提取文本的高维特征, 所以引入了 CNN. 因为基于矩阵变换的 Attention 模型无法对 CNN 抽取的特征进行表征, 所以采用全连接神经网络对 Attention 模型进行改进, 提出了 NN-Attention. 为了加速模型的训练, 采用的循环神经网络为 GRU. 实验采用 CSTSD 数据集, 并用 TensorFlow 完成模型的构建. 实验结果表明, 该模型在 CSTSD 数据集中可以较好地实现文本摘要的自动生成.

关键词: 中文; 短文本; 摘要; GRU; CNN; NN-Attention

引用格式: 何正方, 梁宇. 基于 NN-Attention 的中文短文本摘要. 计算机系统应用, 2020, 29(7): 166-172. <http://www.c-s-a.org.cn/1003-3254/7476.html>

Chinese Short Text Summarization Based on NN-Attention

HE Zheng-Fang, LIANG Yu

(School of Software, Yunnan University, Kunming 650500, China)

Abstract: The Bidirectional RNN (BRNN) was adopted in previous Attention models. The BRNN is effective for context information, but it is unable to extract high dimensional text features. Therefore, the CNN was introduced. The Attention model based on matrix transformation cannot characterize the features extracted by the CNN, a fully-connected neural network is used to improve the Attention model, and the NN-Attention is proposed. The recurrent neural network adopted was GRU, so as to speed up model training, the CSTSD dataset was used and TensorFlow was utilized for model construction. The experimental results show that the proposed model is able to realize automatic generation of text abstracts well in the CSTSD dataset.

Key words: Chinese; short text; summarization; GRU; CNN; NN-Attention

随着 Internet 的飞速发展, 人们越来越多地依赖于万维网来获取所需要的信息. 如何更加有效地浏览和查阅万维网上的海量信息成了当前的研究热点^[1]. 自动文本摘要技术对给定源文本内容进行压缩、提炼与总结, 并产生简洁、流畅且保留关键信息的文本来概括用户所关注的主要内容. 根据摘要内容的来源可以分为抽取式摘要和生成式摘要两类. 其中, 抽取式摘要生成的摘要内容是按一定规则从原文本中抽取出来的词、短语或句子组成的. 生成式摘要产生的摘要文本

则是通过理解文本内容, 用简练的文字将原文本的内容表达出来^[2]. 由于生成式摘要与人类做摘要的过程相似, 所以生成式摘要方法在文本摘要模型中得到了广泛使用.

随着人工智能技术快速发展, 神经网络和深度学习已广泛应用于自然语言处理任务, 比如机器翻译、情感分析、语音识别、文本摘要等. 其中, 以序列到序列 (Sequence2Sequence)^[3]神经网络模型为代表, 其结合了循环神经网络 (Recurrent Neural Network, RNN)^[4]和

① 收稿时间: 2019-12-01; 修改时间: 2019-12-23; 采用时间: 2020-01-02; csa 在线出版时间: 2020-07-03

注意力机制 (attention)^[5], 该模型能够决定需要关注输入的是哪部分, 分配有限的信息处理资源给重要的部分, 比传统的自然语言处理方法效果好很多. 但是以往设计的注意力机制 (BahdanauAttention^[5], LuongAttention^[6]), 不足以将长句的高维特征提取, 会造成导致生成摘要存在句子不通顺、词语重复、可读性差等问题. 这个问题限制了模型的性能, 尤其是当输入序列比较长时, 模型的性能会变得很差, 这个问题在长句的翻译中较为明显.

针对上述问题, 本文重新设计了注意力机制, 以提取长句的特征. 提出了 NN-Attention, 其中的 NN 是全连接神经网络 (Fully-connected Neural Network, FNN)^[7]. 通过实验, 该模型生成的摘要在正确性和可读性上都较以往的注意力模型有所改善.

1 GRU

RNN 是神经网络的一种, 它的输入不仅包括当前时刻的输入还包含了前一个时刻的输出. 基本的 RNN 通过反向传播进行矩阵幂运算, 很容易梯度消失或者梯度爆炸. LSTM^[8] 的提出有效地解决了这个问题, LSTM 通过引入 3 个门控单元, 通过“门”结构来保留重要的内容和去除次要的内容. 但是由于本文引入了 CNN, 参数成几何级增长. 若使用 LSTM, 整个模型的参数量将非常庞大, 训练会变得十分困难. 为了解决 LSTM 参数过多, 训练时间过长, 本文采用 GRU^[9] 神经元. GRU 是 LSTM 的一种变种, 它将遗忘门和输入门合并为一个更新门, 其作用在于每个隐藏层减少了两个矩阵乘法运算, 模型的参数得到减少, 从而减少了训练时间, 并且容易实现, 其示意图如图 1 所示.

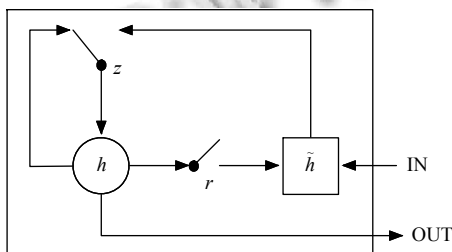


图 1 GRU 示意图

更新门 z 决定隐藏状态的更新是否基于一个新的隐藏状态 \tilde{h} , 重置门 r 状态决定之前的隐藏状态是否被忽略.

让我们来描述 j -th 隐藏单元的激活是如何计算的, 首先, 按照式 (1) 计算重置门 r_j .

$$r_j = \sigma([W_r x]_j + [U_r h_{<t-1>}]_j) \quad (1)$$

式 (1) 中, σ 是 Sigmoid 激活函数, $[\cdot]_j$ 是 j -th 的向量表示. x 和 h_{t-1} 分别表示输入和前一个隐藏状态, W_r 和 U_r 是已经学习到的权重矩阵.

类似的, 按照式 (2) 计算更新门 z_j .

$$z_j = \sigma([W_z x]_j + [U_z h_{<t-1>}]_j) \quad (2)$$

激活隐藏单元 h_j 的实际计算公式如式 (3) 所示.

$$z_j = \sigma([W_z x]_j + [U_z h_{<t-1>}]_j)$$

其中,

$$\tilde{h}_j^{<t>} = \phi([W x]_j + [U (r \odot h_{<t-1>})]_j) \quad (3)$$

在式 (3) 中, 当重置门接近 0 时, 隐藏状态仅重置当前的输入并强制移除之前的状态, 这能有效地阻止无关的信息. 另一方面, 更新门控制多少之前的隐藏状态信息继续留存到当前的隐藏状态, 这和 LSTM 网络中的记忆细胞很像, 并帮助 RNN 记忆长期的信息^[10]. 每一个隐藏单元都有独立的重置门和更新门, 并学习到长时记忆和短时记忆. 隐藏单元因重置门被频繁的激活而学习到短时记忆, 因更新门被频繁激活而学习到长时记忆, 所以 GRU 很适合作为文本摘要模型的神经元.

2 Attention 机制

Sequence2Sequence 模型对输入序列 X 缺乏区分度, 因此引入了 Attention Mechanism 来解决这个问题, 总体模型结构如图 2 所示.

在以往的 Attention 模型中, 只采用了 Bidirectional-RNN^[11], BRNN 对上下文信息是有效的, 但是无法提取文本的高维特征, 所以本文引入了 CNN^[12], CNN 的详细设计见本文的第 3 节, 将 CNN 的输出按照式 (4) 的方式拼接起来.

$$h = \begin{bmatrix} n_i \\ \leftarrow h_i \\ \rightarrow h_i \end{bmatrix} \quad (4)$$

在该模型中, 按照式 (5) 定义了一个条件概率.

$$p(y_i | y_1, \dots, y_{i-1}, x) = g(y_{i-1}, s_i, c_i) \quad (5)$$

其中, s_i 是 Decoder 中 RNN 在 i 时刻的隐状态, 如图 2 所示, 其计算公式如式 (6) 所示.

$$s_i = f(s_{i-1}, y_{i-1}, c_i) \quad (6)$$

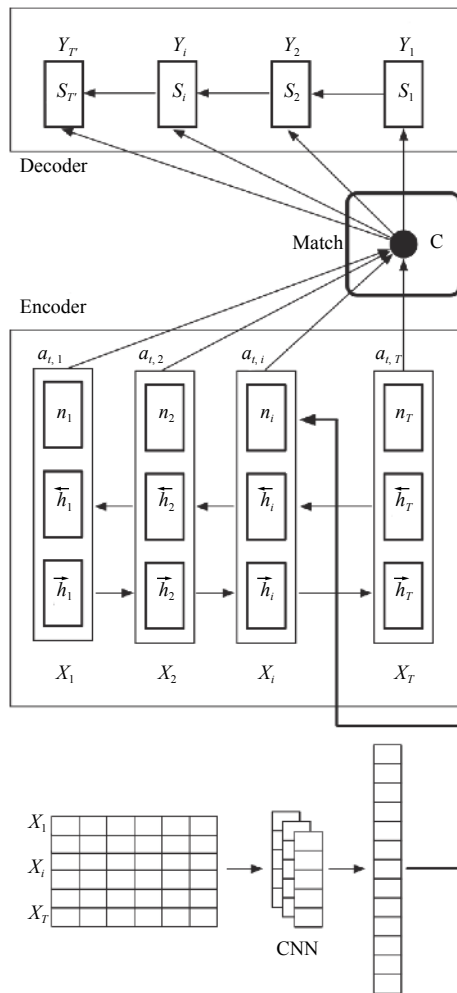


图 2 Attention Mechanism 模型

这里的语义向量 c_i 的计算方式, 与传统的 Sequence2-Sequence 模型直接累加的计算方式不一样, 这里的 c_i 是一个权重化 (weighted) 之后的值, 其表达式如式 (7) 所示.

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j \quad (7)$$

其中, i 表示 Encoder 端的第 i 个词, h_j 表示 Encoder 端的第 j 个词的隐向量, α_{ij} 表示 Encoder 端的第 j 个词与 Decoder 端的第 i 个词之间的权值, 表示源端第 j 个词对目标端第 i 个词的影响程度, α_{ij} 的计算公式如式 (8) 所示.

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

其中,

$$e_{ij} = a(s_{i-1}, h_j) \quad (8)$$

在式 (8) 中, α_{ij} 是一个 Softmax 模型输出, 概率值的和为 1. e_{ij} 表示一个对齐模型, 用于衡量 Encoder 端的位置 j 的词, 对于 Decoder 端的位置 i 的词的齐程度 (影响程度), 换句话说: Decoder 端生成位置 i 的词时, 有多少程度受 Encoder 端的位置 j 的词影响. 以往的对齐模型 e_{ij} 的计算方式如式 (9) 所示^[6]. 为了方便讨论, 对齐模型的表示如图 2 中的 Match 框所示.

$$\text{score}(h_i, \bar{h}_s) = \begin{cases} h_i^T \bar{s}_r \\ h_i^T W_a \bar{s}_r \\ v_a^T \tanh(W_a [h_i; \bar{s}_r]) \end{cases} \quad (9)$$

但是本文在隐藏层增加了 CNN 的输出, 以往的对齐模型已不完全适用, 所以本文应用神经网络重新设计了对齐模型.

3 CNN

本文将卷积神经网络 CNN 应用到中文文本摘要任务, 图 2 中的 CNN 的详细设计如图 3 所示.

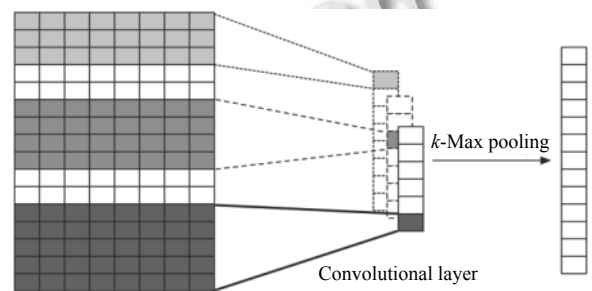


图 3 CNN 模型

CNN 利用 3 个不同 Size 的 Kernel 来提取句子中的关键信息 (类似于多窗口大小的 N -Gram)^[13], 从而能够更好地捕捉局部相关性.

TextCNN 模型最大的缺陷是全局的 Max Pooling 无法提取文本的结构信息, 所以文本中的转折、递进等复杂的语义关系无法被 TextCNN 发现. TextCNN 只能知道文本的关键词是否出现了, 而无法获知关键词出现的顺序和出现的次数. 针对这个问题, 本文采用 k -Max Pooling^[14]做一些优化, k -Max Pooling 针对每个卷

积核都不只保留最大的值,而是保留前 k 个最大值,并且保留这些值出现的顺序,也即按照文本中的位置顺序来排列这 k 个最大值,其在比较复杂的文本上相对于 Max Pooling 会有所提升. k 的取值见表 1.

表 1 不同参数的 TextCNN

参数名称	CNN-1	CNN-2
Word Embedding	128	128
Filter-Size	(3,4,5)	(4,5,6)
CNN_Filters	128	256
k -Max pooling	1	3
是否参数共享	否	是

同时为了更多的提取文本的高维特征,本文增加了 Filter 的数量,具体参数见表 1. 本文为了找到最佳的 Filter-Size 参数,按照表 1 设计了两种 Size 的 Kernel 来训练模型. 此外,为了减少 CNN 的参数,本文对 1 到 T 时刻的 CNN 采用参数共享的方法来训练模型.

为了验证上述的改进对模型精度的影响,本文按照表 1, 分别使用 CNN-1 和 CNN-2 的参数训练模型,并分析生成摘要的质量.

4 NN-Attention

将图 2 中的 Match 框抽象提取出来,如图 4 所示. 其中 h 为 Encoder 的隐藏状态, s 为 Decoder 的隐藏状态, α 为输出的数值.

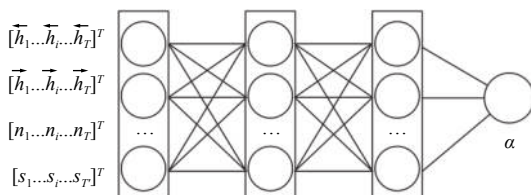


图 4 Match 示意图

因为本文的引入了 CNN, 所以传统的对齐模型已不再适用. 为了提取长句的高维特征, 本文采用神经网络对 Match 进行了重新设计, 具体是全连接神经网络, 如图 5 所示.

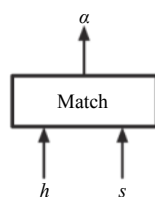


图 5 全连接神经网络示意图

该结构对 x_{n-1} 层和 x_n 层而言, x_{n-1} 层的每一个节点都和第 x_n 层所有节点有连接. 即第 x_n 层的每个节点在进行计算的时候, 激活函数的输入是 x_{n-1} 层所有节点的加权. 由于引入了 CNN, 所以神经网络要按照式 (10) 输入.

$$Input = [h \oplus s]$$

其中,

$$h = [\overleftarrow{h}_1 \cdots \overleftarrow{h}_i \cdots \overleftarrow{h}_T]^T \oplus [\overrightarrow{h}_1 \cdots \overrightarrow{h}_i \cdots \overrightarrow{h}_T]^T \oplus [n_1 \cdots n_i \cdots n_T]^T$$

$$s = [s_1 \cdots s_i \cdots s_T]^T$$
(10)

为了保证 α 为一个数值, 本文的 NN-Attention 最后一层的激活函数使用 Sigmoid 函数^[15]. 为了找到最佳的对齐网络, 本文设计了两层模型, 并用不同的参数训练模型, 具体参数如表 2 所示.

表 2 不同 NN-Attention 的参数

模型编号	第 1 层参数	第 2 层参数
NN-Attention-1	64	64
NN-Attention-2	128	64
NN-Attention-3	128	128

5 实验

5.1 数据集

本文数据集采用的是 CSTSD 数据集^[16], 数据来源于新浪微博主流媒体 (头条新闻、环球网、人民网等) 发布的微博约 68 万数据, 本文按照 8:1:1 的比例切分训练集、验证集、测试集.

5.2 数据预处理

- (1) 去掉无效字符, 如表情、日期、链接、数字等.
- (2) 长度限制, 正文的长度规定为 25 到 250 字, 摘要长度规定为 8 到 35 字.

$$Length_{text} \in (25, 250)$$

$$Length_{label} \in (8, 35) \tag{11}$$

- (3) 相似度过滤, 当正文和摘要有 6 个相同的字时, 本文认为该数据符合条件.

$$Count(T_{text} \cap T_{label}) > 6 \tag{12}$$

5.3 训练

本文引入词嵌入解决矩阵稀疏性问题, 将输入文本映射为 300 维的向量. 首先根据词频统计选取出现频率最高的词汇为 ENCODER 和 DECODER 各定义一个词汇表, 在本文中设置为 60 000, 没在词汇表中的词都转换成一个“Unknown”字符, 然后在训练期间学

习词嵌入的权重。

将新闻和摘要同时输入到 Sequence2Sequence 模型中, 其中新闻对应 Encoder, 摘要对应 Decoder. 整个输入编码和解码的过程中, 按照式 (13) 使用梯度优化算法以及最大似然条件概率为损失函数去进行模型的训练和优化。

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(y_n|x_n) \quad (13)$$

其中, θ 为模型的参数, N 为训练集的样本个数, (y_n, x_n) 是相应的输出和输入的序列^[3]。

本文使用 TensorFlow^[17] 构建模型, 模型的主要参数见表 3。

表 3 模型的主要参数

名称	数值
ENCODER- \vec{h}	256
ENCODER- \tilde{h}	256
ENCODER- n	64
DECODER	576
优化器	AdamOptimization
学习率的初始值	0.005
batch_size	64

5.4 推理——如何生成摘要

当模型训练完成后, 可以得到之前模型没见过的源文章的摘要, 这个过程称为推理. 在推理时, 我们只能访问源句子, 执行解码有很多种方法. 本文采用 Beam Search (束搜索)^[18] 解码, Beam Search 在做摘要时总是将一小部分顶级候选词留在周围, 从而在搜索空间更好地探索所有可能的词汇. Beam 的大小称为宽度, 在本文中设置为 12. 同时本文对生成的摘要的质量采用 ROUGE^[19] 评估中的 ROUGE-1, ROUGE-2 和 ROUGE-L 的 F1 值进行评估。

5.5 采用不同的 TextCNN 参数和不同的 FNN 参数来训练模型

采用如表 1 所示的改进前后两种不同的 TextCNN 参数来训练网络, 模型训练的 Loss 如图 6 所示。

从图 6 可以看出, CNN-2 的 Loss 刚开始下降没有 CNN-1 快, 但是最终的 Loss 比 CNN-1 要小. 训练之后用应用 Beam Search 对测试集的文本进行推理, 并计算 ROUGE 得分, 本文以 Sequence2Sequence 与 Attention 的模型作为基线模型, 计算后的得分如表 4 所示。

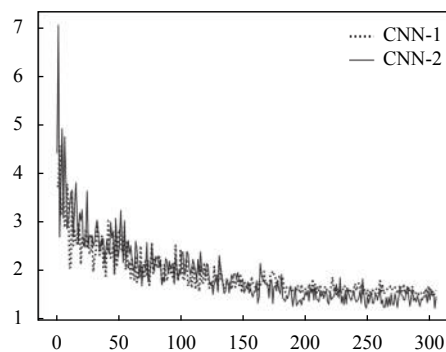


图 6 改进前后的 TextCNN 的 Loss 曲线

表 4 CNN 不同 Size 的参数的 ROUGE F1 得分

模型	1	2	L
Sequence2Sequence+Attention	33.61	17.62	29.53
CNN-1	34.02	17.98	29.87
CNN-2	34.14	18.11	30.02

从表 4 可以看出, 对文本摘要的问题, 在模型引入 CNN 后, 模型的性能有了提升. 模型虽然有了提升, 但是引入 CNN 后, 模型的参数增加很多, 传统的基于矩阵变换的注意力机制已经不能很好地表征这个模型, 所以本文引入了 NN-Attention, 按照表 2 的参数来训练模型, 并使用 CNN-2 的模型来训练, 训练的 Loss 如图 7 所示。

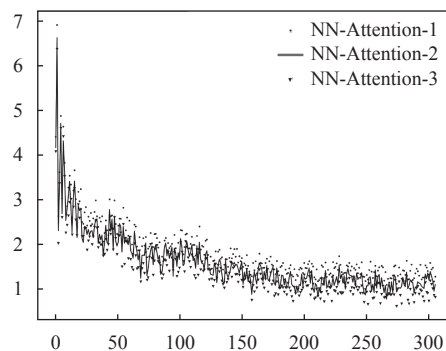


图 7 多种 NN-Attention 参数的 Loss 曲线

训练之后运用 Beam Search 对测试集的文本进行推理, 并计算 ROUGE 得分, 这里以 CNN-2 的模型作为基线模型, 计算后的得分如表 5 所示。

表 5 不同 NN-Attention 的参数的 ROUGE F1 得分

模型	1	2	L
CNN-2	34.14	18.11	30.02
NN-Attention-1	35.16	19.22	31.54
NN-Attention-2	35.47	19.86	31.87
NN-Attention-3	35.33	19.42	32.09

从表5可以看出,对文本摘要的问题,在模型引入 NN-Attention 后,模型的性能有了较大的提升.虽然 NN-Attention-3 模型的损失最小,但是 NN-Attention-2 模型取得了最好的成绩.说明 NN-Attention-2 模型最

适合作为本文提出的 NN-Attention 模型的网络, NN-Attention-3 模型存在过拟合问题.

5.6 案例分析

从测试集前 10 条中随机取 4 条,给出测试结果如表 6.

表 6 测试结果示例

正文	摘要	测试结果
今天上午,陕西省十二届人大常委会第二十二次会议在西安闭幕.会议通过决定:免去郭伯权的陕西省民政厅厅长职务,并同时决定由赵政才接任.今天上午,陕西省十二届人大常委会第二十二次会议在西安闭幕.会议决定免去郭伯权的陕西省民政厅厅长职务.	陕西省人大常委会:免去郭伯权民政厅厅长职务,赵政才接任	免去<Unknown>的陕西省民政厅厅长职务
黄山北站出租车司机说,黄山起步价#元/#公里,价格太低,运营亏本.因本地居民收入不高,黄山出租车已多年未调价.国庆间,从高铁北到景区南大门,##公里要价###,而打表大约###元.黄金周直播秒拍视频####年国庆黄金周期间,央视焦点访谈节目曾曝光过黄山出租车不打表现象.事后,黄山市相关部门开展了出租车行业整治.	黄山出租车司机称“黄金周只议价不打表”,曾被央视曝光	黄山出租车涨价
DATE,从华山北站到游客中心的免费大巴停运.这是因为免费大巴与当地出租车存在利益冲突,部分出租车司机在##日前后采取了围堵景区大巴等过激手段,最终迫使政府部门在##日发文停运免费大巴.黄金周直播视频→部分出租车司机在DATE前后采取了围堵景区大巴等过激手段,最终迫使政府部门在DATE发文取消了免费大巴.	华山免费接站大巴遭拉客出租车围堵,国庆前政府发文责令停运	部分出租车大巴停运
DATE,国庆黄金周的第一天,澎湃新闻记者在甘肃敦煌为您视频直播大漠日出,带“澎友”们体验沙漠日出的壮美.日出预计大概在#点##分出现.不管你身在何方,愿你黄金周有个美好旅程.黄金周直播视频直播入口→DATE,国庆黄金周的第一天,澎湃新闻记者在甘肃敦煌为您视频直播大漠日出,带“澎友”们体验沙漠日出的壮美,开启黄金周的美好旅程.	预告 甘肃敦煌沙漠日出	国庆假期带你体验沙漠日出

6 结论

本文首先对现有 Attention 机制对文本摘要问题研究的缺点做了分析,在以往的 Attention 模型中,只采用了 Bidirectional-RNN, BRNN 对上下文信息是有效的,但是无法提取文本的高维特征,所以本文引入了 CNN.因为本文的引入了 CNN,所以传统的对齐模型已不再适用,本文将 Attention 模型改进,提出了 NN-Attention 以解决这个问题.之后详细描述了 GRU、Attention 机制、CNN、NN-Attention,为了寻找最佳的参数,本文使用多种参数对模型进行训练,通过计算 ROUGE 得分,给出了最佳模型.最后给出了测试结果,从测试结果可以看出,生成的摘要比之前的模型有所进步,但是就准确性与可读性而言,生成的摘要还有待提高,生成式摘要的算法是一个持续性的研究课题.

参考文献

1 胡侠,林晔,王灿,等.自动文本摘要技术综述.情报杂志,2010,29(8):144-147. [doi: 10.3969/j.issn.1002-1965.2010.

08.034]

- 张敏,曾碧卿,韩旭丽,等. DAPC: 结合双注意力和指针覆盖的文本摘要模型. 计算机工程与应用. <http://kns.cnki.net/kcms/detail/11.2127.TP.20190411.1719.018.html>, 2019-12-27.
- Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. Proceedings of the 27th International Conference on Neural Information Processing Systems. Cambridge, MA, USA. 2014.3104-3112.
- Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature, 1986, 323(6088): 533-536. [doi: 10.1038/323533a0]
- Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv: 1409.0473, 2014.
- Luong MT, Pham H, Manning CD. Effective approaches to attention-based neural machine translation. arXiv: 1508.04025, 2015. [doi: 10.18653/v1/D15-1166]
- White BW, Rosenblatt F. Reviewed Work: Principles of Neurodynamics: Perceptrons and the theory of brain mechanisms by Frank Rosenblatt. The American Journal of

- Psychology, 1963, 76(4): 705–707. [doi: [10.2307/1419730](https://doi.org/10.2307/1419730)]
- 8 Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735–1780. [doi: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735)]
- 9 Dey R, Salemt FM. Gate-variants of gated recurrent unit (GRU) neural networks. 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS). IEEE, 2017. 1597–1600.
- 10 Cho K, Van Merriënboer B, Gulcehre C, *et al.* Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv: 1406.1078, 2014. [doi: [10.3115/v1/D14-1179](https://doi.org/10.3115/v1/D14-1179)]
- 11 Schuster M, Paliwal KK. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 1997, 45(11): 2673–2681. [doi: [10.1109/78.650093](https://doi.org/10.1109/78.650093)]
- 12 Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems*. Lake Tahoe, NV, USA. 2012. 1097–1105.
- 13 Kim Y. Convolutional neural networks for sentence classification. arXiv: 1408.5882, 2014. [doi: [10.3115/v1/D14-1181](https://doi.org/10.3115/v1/D14-1181)]
- 14 Shu B, Ren FJ, Bao YW. Investigating Lstm with k-max pooling for text classification. *Proceedings of the 11th International Conference on Intelligent Computation Technology and Automation (ICICTA)*. Changsha, China. 2018. 31–34.
- 15 Han J, Moraga C. The influence of the Sigmoid function parameters on the speed of back propagation learning. *Proceedings of 1995 International Workshop on Artificial Neural Networks*. Malaga-Torremolinos, Spain. 1995. 195–201.
- 16 He ZF. 中文短文本摘要数据集. <https://www.jianshu.com/p/8f52352f0748>. (2018-06-20)[2019-07-01].
- 17 Google. TensorFlow. <https://tensorflow.google.cn/>. [2019-07-11].
- 18 Xiao XEY. Beam Search (集束搜索/束搜索). <https://www.cnblogs.com/xxey/p/4277181.html>. (2019-06-05)[2019-07-11].
- 19 Lin CY. ROUGE: A package for automatic evaluation of summaries. *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*. Barcelona, Spain. 2004. 74–81.