控制与决策
Control and Decision

高速简单循环单元网络

胡枫， 吴义熔， 董方敏， 邹耀斌， 孙水发

引用本文:
胡枫, 吴义熔, 董方敏, 等. 高速简单循环单元网络[J]. 控制与决策, 2022, 37(2): 493−498.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2020.1270

高速简单循环单元网络

胡 枫[1], 吴义熔[1,2], 董方敏[1], 邹耀斌[1], 孙水发[1,2†]

(1. 三峡大学 计算机与信息学院，湖北 宜昌 443002；2. 智慧医疗宜昌市重点实验室，湖北 宜昌 443002)

摘 要: 基于可以并行化计算的简单循环单元(simple recurrent unit, SRU)网络, 引入高速公路网络(highway-networks)的连接思想, 提出高速简单循环单元(H-SRU)网络: 一方面利用非饱和激活函数可以有效缓解梯度消失的性质, 将原有SRU结构里单元状态和隐状态的激活函数替换为非饱和激活函数; 另一方面在SRU的单元状态表示中引入高速公路网络所采用的前馈链接思想, 使网络对梯度变化更敏感; 在此基础上, 基于PTB (penn treebank dataset)和WikiText-2两个数据集构建语言模型, 以验证所提方法的有效性. 实验结果表明, 所设计的高速简单循环单元网络H-SRU在保持SRU原有训练速度优势的同时, 可较大地提高网络的性能. 在WikiText-2数据集上所提方法的困惑度PPL值达到了26.1, 这是目前已知最好效果, 而且其效率也比已知的非SRU网络高.

关键词: 深度学习；循环神经网络；简单循环单元；高速公路网络；激活函数；梯度消失

中图分类号: TP301.6 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2020.1270 **开放科学(资源服务)标识码(OSID):**

引用格式: 胡枫, 吴义熔, 董方敏, 等. 高速简单循环单元网络[J]. 控制与决策, 2022, 37(2): 493-498.

Highway-simple recurrent unit network

HU Feng[1], *WU Yi-rong*[1,2], *DONG Fang-min*[1], *ZOU Yao-bin*[1], *SUN Shui-fa*[1,2†]

(1. College of Computer and Information Technology, China Three Gorges University, Yichang 443002, China;
2. Yichang Key Laboratory of Intelligent Medicine, Yichang 443002, China)

Abstract: Based on the parallelization capability of the simple recurrent unit (SRU) network and the connection strategy of highway-networks, this paper proposes a highway-simple recurrent unit (H-SRU). The H-SRU replaces the activation function of the cell state with the non-saturated activation function to effectively solve the vanishing gradient problems. Additionally, it introduces the idea of feed-forward link used in highway-networks into the cell state representation of the SRU to make the network more sensitive to gradient changes. Natural language processing models are built to verify the effectiveness of the proposed method using the PTB (Penn treebank dataset) and WikiText-2 data sets. The results show that the proposed H-SRU dramatically improves the performance of recognition, while maintaining high training speed. The perplexity value of the H-SRU on the WikiText-2 data set reaches 26.1, which is currently the best known, and its efficiency is higher than that of non-SRU networks.

Keywords: deep learning; recurrent neural networks; simple recurrent unit; highway networks; activation function; vanishing gradient

0 引 言

目前, 基于深度学习的算法已广泛应用于自然语言处理领域. 基于深度学习的语言模型[1]作为自然语言处理中重要的组成部分, 其研究热潮一直高居不下. 其中, 循环神经网络(recurrent neural networks, RNN)具有捕获长序依赖的能力, 被广泛用于语言模型建模中. 但传统循环神经网络会随着网络的加深而出现梯度消失(vanishing gradient)[2]等问题, 于是有学者提出基于长短记忆单元(long short-term memory, LSTM)[3-4]的时间递归神经网络来缓解该问题. 与LSTM类似的门控循环单元(gated recurrent unit, GRU)[5]同样是以词为单位对序列进行循环展开, 然后通过其内部的门限控制信息的输入和输出. 虽然LSTM、GRU的结构能获得更长远的上下文信息, 使模型性能有较大提高, 但其门限繁杂, 使训练速度变得缓慢. 于是近些年来有许多针对LSTM、GRU的改良方案提出, 其中简单循环单元(simple recurrent unit, SRU)[6-7]是最具代表性的改进方案, 它能大大降

收稿日期: 2020-09-12；**录用日期:** 2020-11-23.

基金项目: 国家自然科学基金项目(61871258)；NSFC-新疆联合基金重点项目(U1703261)；国家重点研发计划项目(2016YFB0800403).

†通讯作者. E-mail: watersun@ctgu.edu.cn.

低神经网络训练耗时. SRU训练速度较快的原因在于,它的门限舍弃了循环神经单元的时间参数[6],但也因此导致精度降低.

深度学习的成功主要归因于它的深层结构[8]. 神经网络的结构层次越深往往获取到的特征信息越丰富,特别是卷积神经网络,网络层数的增加会使模型性能增强[9-10]. 然而,事实上随着网络层数的增加,深层网络的优化是相当困难的[11]. 目前已提出多种深层前馈神经网络结构,其中比较具有代表性的有高速公路网络(highway-networks)[8,12],以及用于CNN的残差网络(residual-networks)[13]. 高速公路网络和残差网络两者结构相似,都用于解决深度神经网络易产生梯度消失以及网络退化的训练问题,但highway-networks方法会增加网络参数量,且训练精度较差[13]. 从形式上,residual-networks可以理解为highway-networks的简化版.

本文通过对SRU结构的深入研究发现,修改其激活函数并将高速公路网络的连接思想引入单元状态中,可以使SRU对梯度变化更敏感;而对于使用非饱和激活函数带来的梯度爆炸隐患,本文将采用批标准化(batch normalization, BN)[14]的方法应对. 本文的实验主要是将所提方法应用于语言模型的构建,结果表明,本文提出的H-SRU可取得比LSTM、GRU、SRU等网络更好的效果,在训练耗时方面比非SRU网络(比如LSTM、GRU)更低.

1 现有工作介绍

1.1 简单循环单元

简单循环单元(SRU)是LSTM的一种变体,它们的共同点是每个神经元都是一个处理单元,每个处理单元里都含有若干门限,门限用于控制信息流. SRU单元结构如图1所示,公式定义如下:

$$f_t = (\sigma(W_f x_t + b_f)), \tag{1}$$

$$r_t = (\sigma(W_r x_t + b_r)), \tag{2}$$

$$c_t = (f_t \odot c_{t-1} + (1 - f_t) \odot W x_t), \tag{3}$$

$$h_t = (r_t \odot \mathrm{Tanh}(c_t) + (1 - r_t) \odot x_t). \tag{4}$$

其中:x_t为当前层的t时刻的输入值;f_t和r_t分别为t时刻的遗忘门和重置门,遗忘门和重置门的激活函数σ都是Sigmoid函数;c_{t-1}为$t-1$时刻的单元状态输出值,c_t为t时刻的单元状态;h_t为时刻t的隐状态向量;Tanh为隐状态的双曲正切激活函数(hyperbolic tangent activation function, Tanh);W、W_f、W_r为模型权重参数;b_f、b_r为偏置向量.

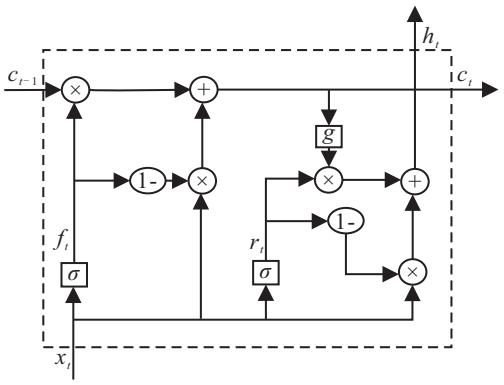

图1 SRU结构

与LSTM和GRU等循环神经网络不同的是,SRU解除了对前一时刻隐藏层输出结果的依赖性. 从式(1)和(2)可以看出,SRU网络已经解除了对h_{t-1}的依赖,这样可以实现程序的并行化处理,从而能非常迅速和简洁地进行计算. 而Transformer[15]结构的前馈层同样没有这些依赖关系,因此在前馈层传递信息时可以并行执行各种路径.

1.2 高速公路网络

高速公路网络[12]针对的是神经网络中层数过高,且没有特殊限制而产生冗余网络层的问题. 它的体系结构特点是使用门控单元,通过网络学习调节信息流. 高速公路网络以前一层输出乘以门限T加上当前层的输出乘以门限C作为新的当前层的输出. 常用的公路层公式定义如下:

$$T(x) = (\sigma(Wx + b)), \tag{5}$$

$$C(x) = (1 - T(x)), \tag{6}$$

$$H(x) = (T(x) \odot F(x) + C(x) \odot x). \tag{7}$$

其中:x为输入,$F(x)$为原始的输出,W为权重,b为偏置向量,$C(x)$和$T(x)$为网络的门控,σ为激活函数,$H(x)$为高速公路网络的输出. 从上述内容中可以看出:当$T(x) = 1$时,新的输出$H(x)$与$F(x)$为恒等映射关系;当$T(x) = 0$时,新的输出$H(x)$由上一层的输入信息x构成. 常用公路层结构如图2所示.

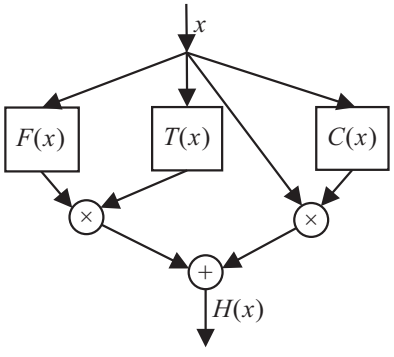

图2 高速公路网络结构

1.3 激活函数

1) 饱和激活函数.

饱和激活函数的缺点是如果输入x非常大或非常小,则对应的梯度可能就很小,这样就会使梯度下降算法效率变低. Tanh函数是一个典型的饱和激活函数,它的函数如下所示:

$$\mathrm{Tanh}(x)=\frac{\mathrm{e}^{x}-\mathrm{e}^{-x}}{\mathrm{e}^{x}+\mathrm{e}^{-x}}. \tag{8}$$

2) 非饱和激活函数.

由于非饱和激活函数的性质,神经网络学习速度通常会快很多,同时能防止梯度消失的问题. 典型的非饱和激活函数有线性整流函数(rectified linear unit, ReLU)[16],其函数如下所示:

$$\mathrm{ReLU}(x)=\max(0,x). \tag{9}$$

2 高速简单循环单元

2.1 高速简单循环单元结构

针对循环神经网络的梯度消失、梯度爆炸和堆叠网络层数会出现的网络退化[8],以及网络训练速度慢等问题,本文通过改进SRU结构来解决. 在SRU的算法公式里,单元状态公式是比较核心的公式,它关系着隐状态向量的输出,也就是式(3),所以本文针对单元状态公式进行改进,主要分为如下3个方面:

1) 改变激活函数.

文献[16]提出了在循环神经网络中使用ReLU作为激活函数的想法. 本文将SRU的隐状态的激活函数改为非饱和激活函数ReLU,从而避免由饱和激活函数Tanh引起的梯度消失问题,同时也能加快网络收敛速度. 激活函数也可以被替换为其他非饱和激活函数,选择ReLU是因为它具有代表性.

2) 添加高速连接.

该结构是受高速公路网络的启发来调节信息流,改进的主要方式是参考highway connection的连接,将其连接方法引入SRU的单元状态中,即式(3)中. 对于引入的信息,本文使用的是前一层激活的单元状态c_t^{l-1},而不是前一层的全部输出信息x_t^{l-1},原因是本文认为上一层的最终输出比单元状态的输出具备更多的冗余信息,且只引入前一层单元状态的输出信息c_t^{l-1}将更利于当前层单元状态的计算. 改进后的结构:由前一层激活的单元状态输出信息c_t^{l-1}与遗忘门的耦合版本之积,同当前层的单元状态的输出信息c_t^l与遗忘门之积,上述两部分之和,作为新的当前层的单元状态的输出信息M_t^l,输入到隐状态中进行下一步计算. 改进后的单元状态公式如下:

$$M_t^l=(f_t^l\odot c_t^l+(1-f_t^l)\odot g(c_t^{l-1})), \tag{10}$$

$$h_t^l=(r_t^l\odot g(M_t^l)+(1-r_t^l)\odot x_t^l). \tag{11}$$

其中:l为堆叠层数,c_t^l为l层t时刻原始的单元状态输出,M_t^l为l层t时刻新的单元状态输出,c_t^{l-1}为$l-1$层t时刻的单元状态向量,h_t^l为l层t时刻的输出状态,g为ReLU激活函数.

3) 运用批标准化.

批标准化[14]是针对每层训练小批量的预激活均值和方差进行规范化,以此来解决数据内部协变量偏移(internal covariate shift)的问题,加速训练的同时也能在一定程度上避免过拟合. 在SRU结构中加入批标准化,主要是考虑到改变其激活函数后可以有效缓解由非饱和激活函数造成的梯度爆炸问题,而且文献[13,17]也表明,将非饱和激活函数与批标准化技术相结合有不错的效果. 结合批标准化之后,本文设计的H-SRU第l层的单元结构公式如下:

$$f_t^l=(\sigma\mathrm{BN}(W_f^l x_t^l)), \tag{12}$$

$$r_t^l=(\sigma\mathrm{BN}(W_r^l x_t^l)), \tag{13}$$

$$c_t^l=(f_t^l\odot c_{t-1}^l+(1-f_t^l)\odot\mathrm{BN}(W^l x_t^l)), \tag{14}$$

$$M_t^l=(f_t^l\odot c_t^l+(1-f_t^l)\odot\mathrm{ReLU}(c_t^{l-1})), \tag{15}$$

$$h_t^l=(r_t^l\odot\mathrm{ReLU}(M_t^l)+(1-r_t^l)\odot x_t^l). \tag{16}$$

其中:BN表示引入批标准化,ReLU为激活函数,M_t^l为引入的前馈高速公路链接单元.

根据式(12)～(16)画出H-SRU结构示意图,如图3所示,其中虚线表示引入高速公路网络前馈链接的信息流通的方向. 由虚线指引,当前层c_t^l状态信息与遗忘门f_t之积,及被激活的前一层c_t^{l-1}状态信息与f_t的耦合版本之积,两部分相加即可得到新的输出M_t^l. 之后,将被激活的M_t^l输入到隐藏层状态向量中进行下一步的循环计算.

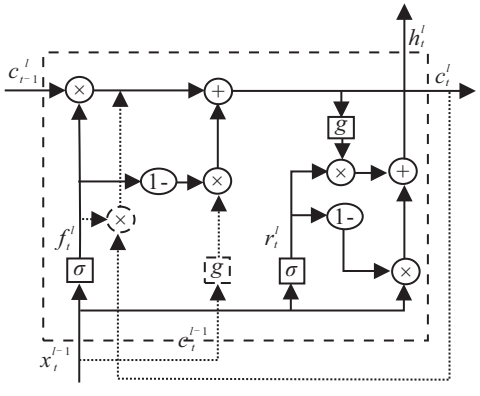

图3　H-SRU结构

批标准化的性质在于消除偏差,所以在式(12)、(13)中偏置向量b被忽略.

2.2 高速简单循环单元结构的反向传播

一般的循环神经网络,误差项的反向传播(back propagation, BP)包括两个方向:一个是沿时间的反向传播(backpropagation through time, BPTT),即从t时刻起,计算每个时刻的误差项;另一个则是将误差在空间上向上一层传播.本文设计的SRU结构是基于神经网络空间结构建立的,所以仅对添加的连接项即单元状态的BP展开讨论.

式(14)、(15)为H-SRU的单元状态公式,为了计算方便省略$1-f_t^l$,得到

$$M_t^l = f_t^l \odot (f_t^l \odot c_{t-1}^l + W^l x_t^l) + \mathrm{ReLU}(c_t^{l-1}). \quad (17)$$

利用$F(c)$对式(17)中的部分函数进行等效替换,可得

$$F(c) = (f_t \odot (f_t \odot c_{t-1} + W x_t)). \quad (18)$$

假设当前状态时刻为t,设l为计算的误差项$\delta_{c,t}^l$所在层数,L为当前网络层数,L层到l层之间有若干个网络层. 由于ReLU激活函数的性质,当c_t^{L-1}大于0时,式(17)可以简化为

$$M_t^L = f_t^L \odot (f_t^L \odot c_{t-1}^L + W^L x_t^L) + \mathrm{ReLU}(c_t^{L-1}) = F(c) + \mathrm{ReLU}(c_t^{L-1}) = c_t^l + \sum_{n=l}^{L-1} F(c)^n. \quad (19)$$

因此对M_t^L求c_t^l的偏导,可以展开为

$$\frac{\partial M_t^L}{\partial c_t^l} = \frac{\partial\Big(c_t^l + \sum\limits_{n=l}^{L-1} F(c)^n\Big)}{\partial c_t^l} = 1 + \frac{\partial\Big(\sum\limits_{n=l}^{L-1} F(c)^n\Big)}{\partial c_t^l}. \quad (20)$$

故结合ReLU激活函数的性质以及链式求导法则,得出单元状态的第l层的误差项$\delta_{c,t}^l$为

$$\delta_{c,t}^l = \frac{\partial E}{\partial c_t^l} = \frac{\partial E}{\partial h_t^L}\frac{\partial h_t^L}{\partial M_t^L}\frac{\partial M_t^L}{\partial c_t^l} = \frac{\partial E}{\partial h_t^L}\Big(1 + \frac{\partial\Big(\sum\limits_{n=l}^{L-1} F(c)^n\Big)}{\partial c_t^l}\Big) r_t^L. \quad (21)$$

其中:E为误差,h_t^L为L层t时刻的隐状态向量,r_t^L为L层重置门限. 由式(21)可知,对SRU结构改变其激活函数的同时,将高速连接表达式引入其单元状态公式定义中,推导出的反向传播误差项δ中能提取出单位矩阵$\mathbf{1}$,在一定程度上能避免链式求导法则中连续相乘使偏导数趋近于0而引起的梯度消失问题.

3 实验和结果分析

3.1 数据集和配置

本文实验中的模型训练使用的神经网络都是在Linux系统上基于Pytorch平台搭建的,机器配置使用NVIDIA GeForce RTX 2080TI显卡进行加速训练. 本文将从语言模型建模的角度出发,使用PTB数据集和WikiText-2数据集. 为了实验的公平性,本文实验中的每种循环神经网络结构都设置相同的参数.

3.2 PTB语言模型建模

本文测试H-SRU结构在语言模型建模方面的能力,使用的数据集是PTB[18],数据集中包含了9 998个不同的单词词汇,加上稀有词语的特殊符号和语句结束标记符,一共是10 000个词汇. 完成PTB数据集训练和测试的源码是基于Pytorch官方的语言模型示例. 在此数据集上本文使用transformer、LSTM、GRU、R-GRU以及SRU进行对比实验. 为了更好地比较各个网络的优劣,实验中用到的循环神经网络参数设置都是相同的:隐藏层均设置650个神经元,Embeddings的大小均设置为650,丢弃率(drop out)的大小均设置为50 %. 以层数堆叠的方式训练神经网络,每种神经网络分别进行3、5、7层的网络训练,本文实验将采用困惑度(perplexity, PPL)作为评判标准[1],具体实验结果如表1所示.

表1 不同网络测试PTB数据集的结果

NET	3 layer		5 layer		7 layer	
	PPL	t/s	PPL	t/s	PPL	t/s
transformer[15]	138.76	**34.26**	129.00	**53.16**	124.17	**71.89**
LSTM[4]	59.68	286.53	77.17	397.25	68.65	552.96
GRU[5]	**57.97**	239.40	52.45	315.97	63.04	436.58
R-GRU[19]	90.51	236.15	49.69	347.04	*54.35*	446.84
SRU[6]	86.01	155.60	68.49	*243.34*	55.39	279.32
relu+SRU	102.20	166.50	98.04	316.18	88.26	393.57
highway+SRU	82.55	215.79	73.68	320.69	60.22	423.61
H-SRU	61.85	186.22	**44.41**	263.80	**42.66**	358.72

表1的实验数据表明,transformer结构具有较快的运行速度,但准确率却极低. 传统循环神经网络(LSTM、GRU)在3层时都有不错的性能,但是随着网络层数的升高,由于梯度消失等问题,性能都有所变差,其中LSTM在5层的性能比其他层更差,可能是此时的梯度下降算法使梯度到达了局部最优点而不是全局最优点. R-GRU[19]是基于GRU添加残差

信息的网络结构,这种结构通过建立层与层之间的连接在一定程度上也能有效针对梯度消失问题. 从表1可以看出,R-GRU在层数为3时性能不如其他网络,在加深网络层数后,其性能不断提升,在对应层上要优于GRU. 由于SRU具有并行化的计算能力,训练耗时在所有的网络结构中都是较快的,但也因为省去了一部分的信息导致网络在语言模型预测单词方面并没有优势. 本文设计的H-SRU在对应的层数上其网络性能要优于其他网络,虽然在添加一些信息后训练速度相比于原本的SRU略慢一些,但是预测词语的能力有所提高,能有效降低PPL值. 值得一提的是,本文在SRU结构的基础上针对只修改激活函数的(relu+SRU)以及只引入高速公路的前馈链接(highway+SRU)进行了消融实验,结果表明,本文提出的H-SRU都有较明显的优势. 另外,层数堆叠使模型的参数随之增多,网络的训练速度会变得越来越慢.

通过表1的数据可以发现,本文设计的H-SRU结构继承了原本SRU训练速度快的优势,同时也有着较优的PPL值,而且在深层网络依然能保持良好的效果. 本文统计了SRU和H-SRU在训练过程中迭代计算的损失值,绘制出其损失函数变化曲线如图4所示.

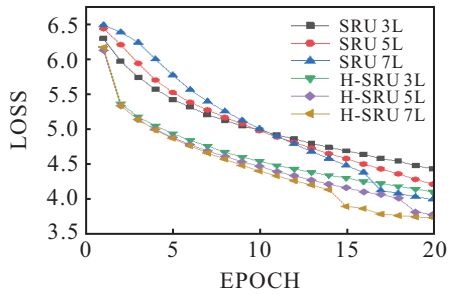

图4　H-SRU和SRU在PTB数据集上的损失值变化

通过对比损失函数变化曲线可以明显地看出,本文设计的H-SRU在迭代过程中损失值下降更快. 分析可知,正是因为本文设计的H-SRU在SRU单元状态里添加前一层的单元状态信息,使网络对梯度变化更敏感,反向传播在层与层之间能传递有效信息. 表1的数据则更直观地表明,本文设计的H-SRU相比于传统神经网络(LSTM、GRU)在性能上都有较大的提升,且在堆叠层数后依然有较好的模型效果. 而且表1数据表明,在网络训练速度上,本文设计的H-SRU也比LSTM、GRU、R-GRU快很多.

3.3　WikiText-2语言模型建模

为了进一步验证本文设计的网络结构在语言模型建模上的有效性,采用与PTB类似的WikiText-2数据集进行实验,WikiText-2[20]比PTB数据集要大一倍多. 继续选用transformer、LSTM、GRU、R-GRU、SRU以及本文设计的H-SRU作对比实验. 网络训练参数配置延续之前的参数. 为了节省训练时间,只进行7层网络的实验,具体结果如表2所示.

表2　WikiText-2数据集在各个网络测试结果

NET	layers	PPL	t/s
transformer[15]	7	161.43	**166.81**
LSTM[4]	7	55.18	1 226.06
GRU[5]	7	57.17	983.43
R-GRU[19]	7	28.45	1 050.83
SRU[6]	7	55.24	636.62
relu+SRU	7	89.05	751.19
highway+SRU	7	<u>54.02</u>	1 179.72
H-SRU	7	**26.10**	<u>861.98</u>

从表2的数据中可以看出,本文设计的H-SRU依然能有效降低PPL值,并且网络训练速度仅次于原始SRU以及transformer结构. 值得一提的是,R-GRU在此数据集上表现较好,在降低PPL值方面与H-SRU效果相差无几,但是本文设计的H-SRU在网络训练速度上要快于R-GRU. 通过表1和表2的数据对比可以发现,各个网络在两个数据集上的实验效果基本类似,这也验证了本文改进的有效性. 另外,本文也开展了在SRU结构里加入残差信息的实验[19],虽然在深层网络(9层)里有一定的效果,但是效果并不明显,这意味着残差信息并不适用于SRU这类具有高并行性的网络.

4　结　论

在循环神经网络结构中,反向传播信息更新梯度极易发生梯度消失的问题,且随着堆叠层数的增加,网络退化问题也尤为严重,这些问题都将使得网络在构建语言模型时性能恶化,导致模型预测词语的能力急剧下降. 本文基于SRU提出的H-SRU结构继承了原始SRU训练速度快的优势,同时能有效缓解深层网络中的梯度消失等问题. 本文在PTB和WikiText-2两个数据集上进行了实验,实验结果表明,无论是在训练速度还是准确性方面,H-SRU与传统神经网络相比都有较大的提升,即使增加网络深度,其依然有较好的模型效果. 另外,当使用残差连接时,基于GRU的模型R-GRU也能缓解梯度消失的问题,但是在对应的层数上,本文设计的H-SRU模型与R-GRU模型同样有显著的优势. 需要说明的是,本文设计的H-SRU不能避免过拟合现象,在特别深的网络中,模型效果变差,不能有效降低PPL值,后续将尝试做进

一步修改,比如优化网络结构复杂度,更好地利用深层结构进一步提高模型性能.

参考文献(References)

[1] Mikolov T, Zweig G. Context dependent recurrent neural network language model[C]. Proceedings of the 2012 IEEE Spoken Language Technology Workshop. Miami, 2012: 234-239.

[2] Hochreiter S. The vanishing gradient problem during learning recurrent neural nets and problem solutions[J]. International Journal of Uncertainty, Fuzziness and Knowledge Based Systems, 1998, 6(2): 107-116.

[3] 龙小强, 李捷, 陈彦如. 基于深度学习的城市轨道交通短时客流量预测[J]. 控制与决策, 2019, 34(8): 1589-1600.
(Long X Q, Li J, Chen Y R. Metro short-term trac ow prediction with deep learning[J]. Control and Decision, 2019, 34(8): 1589-1600.)

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

[5] Cho K, Van Merrienboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[J]. 2014, arxiv: 1406.1078.

[6] Lei T, Zhang Y, Wang S I, et al. Simple recurrent units for highly parallelizable recurrence[C]. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. Brussels: ACL, 2018: 4470-4481.

[7] 张文, 冯洋, 刘群. 基于简单循环单元的深层神经网络机器翻译模型[J]. 中文信息学报, 2018, 32(10): 36-44.
(Zhang W, Feng Y, Liu Q. Deep neural machine translation model based on simple recurrent units[J]. Journal of Chinese Information Processing, 2018, 32(10): 36-44.)

[8] Srivastava R K, Greff K, Schmidhuber J. Training very deep networks[C]. Proceedings of the 28th International Conference on Neural Information Processing Systems. Montreal: NeurIPS, 2015: 2377-2385.

[9] Gehring J, Auli M, Grangier D, et al. Convolutional sequence to sequence learning[C]. Proceeding of the 34th International Conference on Machine Learning. Sydney: PMLR, 2017: 1243-1252.

[10] Conneau A, Schwenk H, Barranlt L, et al. Very deep convolutional networks for text classification[C]. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Valencia: ACL, 2017: 1107-1116.

[11] Lee C Y, Xie S, Gallagher P, et al. Deeply-supervised nets[C]. Proceedings of the 18th International Conference on Artificial Intelligence and Statistics. San Diego: JMLR, 2015: 562-570.

[12] Srivastava R K, Greff K, Schmidhuber J, et al. Highway networks[J]. 2015, arxiv: 1505.00387.

[13] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, 2016: 770-778.

[14] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]. Proceedings of the 32nd International Conference on Machine Learning. Lille: IMLS, 2015: 448-456.

[15] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. 2017, arxiv: 1706.03762.

[16] Quoc V L, Navdeep J, Geoffrey E H. A simple way to initialize recurrent networks of rectified linear units[J]. 2015, arxiv: 1504.00941.

[17] Ravanelli M, Brakel P, Omologo M, et al. Light gated recurrent units for speech recognition[J]. IEEE Transactions on Emerging Topics in Computational Intelligence, 2018, 2(2): 92-102.

[18] Marcus M P, Marcinkiewicz M A, Santorini B. Building a large annotated corpus of English: The penn treebank[J]. Computational Linguistics, 2002, 19(2): 313-330.

[19] 张忠豪, 董方敏, 孙水发, 等. 基于残差的门控循环单元[J]. 自动化学报, DOI: 10.16383/j.aas.c190591.
(Zhang Z H, Dong F M, Sun S F, et al. Residual based gated recurrent Unit[JL]. Acta Automatica Sinica, DOI: 10.16383/j.aas.c190591.)

[20] Melis G, Dyer C, Blunsom P. On the state of the art of evaluation in neural language models[J]. 2017, arxiv: 1707.05589.

作者简介

胡枫(1994−), 男, 硕士生, 从事人工智能和自然语言处理的研究, E-mail: hfeng0011@gmail.com;

吴义熔(1970−), 男, 教授, 博士生导师, 从事人工智能和自然语言处理等研究, E-mail: yirongwu@gmail.com;

董方敏(1965−), 男, 教授, 博士生导师, 从事计算图形学、计算机视觉和人工智能、CT图像重建等研究, E-mail: fmdong@ctgu.edu.cn;

邹耀斌(1978−), 男, 副教授, 博士, 从事数字图像处理和模式识别等研究, E-mail: zyb@ctgu.edu.cn;

孙水发(1977−), 男, 教授, 博士生导师, 从事多媒体信息处理、智能信息处理等研究, E-mail: watersun@ctgu.edu.cn.

(责任编辑：闫 妍)