Conformer: Convolution-augmented Transformer for Speech Recognition

Conformer：用于语音识别的卷积增强Transformer

*Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo*

*Wang, Zhengdong Zhang, Yonghui Wu, Ruoming Pang*

Google Inc.

# 摘要

Recently Transformer and Convolution neural network (CNN) based models have shown promising results in Automatic Speech Recognition (ASR), outperforming Recurrent neural networks (RNNs). Transformer models are good at capturing content-based global interactions, while CNNs exploit local features effectively. In this work, we achieve the best of both worlds by studying how to combine convolution neural networks and transformers to model both local and global dependencies of an audio sequence in a parameter-efficient way. To this regard, we propose the convolution-augmented transformer for speech recognition, named Conformer. Conformer significantly outperforms the previous Transformer and CNN based models achieving state-of-the-art accuracies. On the widely used LibriSpeech benchmark, our model achieves WER of 2.1%/4.3% without using a language model and 1.9%/3.9% with an external language model on test/testother. We also observe competitive performance of 2.7%/6.3% with a small model of only 10M parameters.

**Index Terms**: speech recognition, attention, convolutional neural networks, transformer, end-to-end

近年来，基于Transformer和卷积神经网络（CNN）的模型在自动语音识别（ASR）中取得了很好的效果，优于递归神经网络（RNNs）。Transformer模型擅长捕获基于内容的全局交互，而CNN则有效地利用了局部特征。在这项工作中，我们通过研究如何结合卷积神经网络和Transformer，以参数有效的方式对音频序列的局部和全局依赖进行建模，从而达到两全其美的效果。为此，我们提出了一种用于语音识别的卷积增强变换器Conformer。Conformer显著优于以前基于Transformer和CNN的模型，取得了目前最好的准确度。在广泛使用的LibriSpeech基准上，对于test/testother测试集，我们的模型在不使用语言模型的情况下达到2.1%/4.3%字错误率，在使用外部语言模型时达到1.9%/3.9%字错误率。我们还注意到只有10M参数的小型模型的WER为2.7%/6.3%，颇具竞争性。

**关键词**：语音识别，注意力，卷积神经网络，transformer，端到端

# 1.     前言

基于神经网络的端到端自动语音识别（ASR）系统近年来有了很大的发展。递归神经网络（RNN）已经成为ASR的实际选择[1，2，3，4]，因为它们可以有效地模拟音频序列中的时间依赖性[5]。最近，基于self-attention的Transformer体系结构[6,7]由于能够捕获长距离交互和高训练效率而被广泛用于序列建模。同时，卷积也成功地用于ASR[8，9，10，11，12]，它通过局部感受野逐层渐进地捕捉局部上下文。

然而，使用self-attention或卷积的模型都有其局限性。虽然transformer擅长对远程全局上下文进行建模，但它们提取细粒度局部特征模式的能力较差。另一方面，卷积神经网络（CNNs），作为视觉（计算）中实际计算模块，善于提取局部信息。CNN通过一个局部窗口学习共享的基于位置的kernels，该窗口保持了转换的等价性，并且能够捕获边缘和形状等特征。使用局部连接的一个限制是需要更多的层或参数来捕获全局信息。为了解决这个问题，论文ContextNet[10[[1]](#footnote-1)]在每个残差块中采用了压缩和激励（squeeze-and-excitation）模块[13]，以捕获更长的上下文。然而，它在捕获动态全局上下文方面仍然受到限制，因为它对整个序列仅应用全局平均。

最近的工作表明，组合卷积和self-attention比单独使用它们更有效[14[[2]](#footnote-2)]。他们能够共同学习位置相关的本地特性，并使用基于内容的全局交互。同时地，像[15,16]这样的论文利用相对位置信息增强self-attention，保持了不变性。Wu等人[17[[3]](#footnote-3)]提出了一种多分支架构，将输入分成两个分支：self-attention和卷积；并将它们的输出串联起来。他们的工作以移动应用为目标，并显示了在机器翻译任务上的改进。

在这项工作中，我们研究如何在ASR模型中有机地结合卷积与self-attention。我们假设全局和局部相互作用对参数有效性都很重要。为了实现这一点，我们提出了一种新的self-attention和卷积的组合，它将实现两个方面的最佳效果——self-attention学习全局交互作用，而卷积有效地捕获基于相对偏移的局部相关性。受Wu等人[17，18]的启发，我们引入了一种新的self-attention和卷积的组合，夹在一对前馈模块之间，如图1所示。

我们提出的模型，命名为Conformer，在LibriSpeech上实现了最优的结果。当使用外部语言模型时，在testother数据集上，比以前公开发布的最好的Transformer Transducer[7[[4]](#footnote-4)]的性能要好15%（相对改进）。基于10M、30M和118M的模型参数限制条件，我们提出了三个模型。我们的10M模型在test/testother数据集上的结果是2.7%/6.3%，比同类规模的当代工作[10]有提高。我们的中型30M参数模型已经优于[7]中发布的139M参数的transformer transducer模型。使用118M参数模型，我们可以在不使用语言模型的情况下WER达到2.1%/4.3%，在使用外部语言模型的情况下达到1.9%/3.9%。

我们进一步仔细研究了attention head的数量、卷积核的大小、激活函数、前馈层的位置以及在基于transformer的网络中添加卷积模块等不同策略的影响，并阐明了每种策略如何有助于提高精确度。

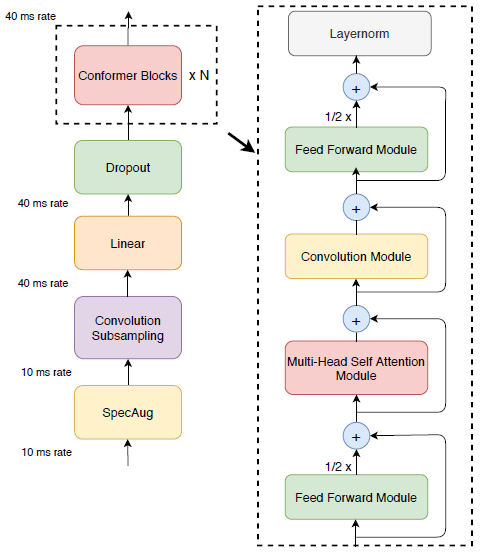


图1：Conformer编码器模型结构。两个macaron样式的前馈层中间夹着self-attention和卷积模块，后接layernorm。

|  |
| --- |
| 图2：卷积模块。卷积模块包含一个扩展因子为2的逐点卷积，该扩展因子通过GLU激活层投射通道数，然后是一个一维纵深卷积。1-D depthwise conv后面是Batchnorm和swish激活函数。 |

# 2.     Conformer Encoder

我们的语音编码器首先使用卷积子采样层处理输入，然后使用多个conformer块，如图1所示。相较与[7,19]，我们模型的显著特点是使用conformer代替transformer。

Conformer块由四个模块叠加而成，即前馈模块、self-attention模块、卷积模块和最后的第二个前馈模块。在2.1、2.2和2.3节分别介绍了self-attention、卷积和前馈模块。最后，2.4描述了如何组合这些子块。

## 2.1.    多头self-attention模块

我们采用了multi-headed self-attention（MHSA），同时集成了Transformer XL[20[[5]](#footnote-5)]的一项重要技术，即相对正弦位置编码方案。相对位置编码使得self-attention模块在不同的输入长度上具有更好的泛化能力，并且使编码器对语句长度的变化具有更强的鲁棒性。我们使用带dropout的pre-norm残差单元[21，22]，这有助于训练更深层次的模型。下面的图3说明了多头self-attention块。

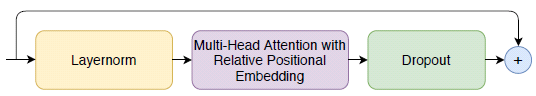


图3：多头self-attention模块。我们使用多头self-attention与相对位置嵌入在一个pre-norm residual unit中。

## 2.2.    卷积模块

受[17]的启发，卷积模块从一个gating mechanism开始[23]——一个pointwise卷积和一个gated linear unit（GLU）。然后是一个一维纵深卷积层。Batchnorm部署在卷积之后，以帮助训练深度模型。图2展示了卷积模块。

## 2.3.    前馈模块

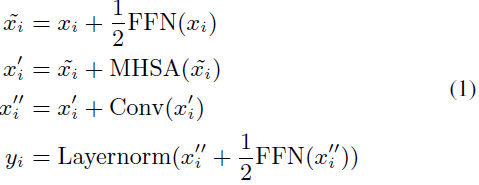
[6]中提出的transformer结构在MHSA层之后部署了一个前馈模块，由两个线性变换和中间的一个非线性激活组成。在前馈层上添加残差连接，然后进行层规范化。这种结构也被[7，24]所采用。

我们仍遵循pre-norm残差单元[21，22]，并在残差单元内和第一个线性层之前的输入上应用层归一化。我们还应用了Swish激活函数[25]和dropout，这有助于正则化网络。图4说明了前馈（FFN）模块。

## 2.4.    Conformer Block

我们提出的conformer block包含两个前馈模块，中间夹着多头self-attention模块和卷积模块，如图1所示。

这种三明治结构的灵感来自Macaron Net[18[[6]](#footnote-6)]，它建议将transformer块中的原始前馈层替换为两个半步前馈层，一个在注意层之前，一个在注意层之后。像Macron网络中一样，我们在前馈（FFN）模块中使用了半步残差。第二个前馈模块后面是最后一个layernorm层。在数学上，这意味着，对于conformer块的输入，该块的输出是：



其中，FFN是指前馈模块，MHSA是指多头self-attention模块，Conv是指卷积模块，如前几节所述。

我们在第3.4.3节中讨论的对比研究，比较了Macaron风格的半步FFN和以前工作中使用的FFN。我们发现，在我们的conformer架构中，有两个Macaron网络式的前馈层，其中有半步残差连接，将注意力和卷积模块夹在中间，比有一个前馈模块有显著的改进。

|  |
| --- |
| 图4：前馈模块。第一个线性层使用4的展开因子，第二个线性层将其投影回模型维度。我们在前馈模块中使用了swish激活函数和pre-norm残差单元。 |

卷积和self-attention的结合已经被研究过了，人们可以想象很多方法来实现这一点。第3.4.2节研究了self-attention与卷积的不同结合方法。我们发现卷积模块堆叠在self-attention模块之后，最适合于语音识别。

# 3.     实验

## 3.1.    数据

我们在LibriSpeech[26]数据集上评估了该模型，该数据集由970小时的标记语音和额外的800M单词标记文本语料库组成，用于建立语言模型。我们提取了80个通道滤波器组的特征，这些特征是从一个25ms的窗口以10ms的步幅计算出来的。我们使用SpecAugment[27，28]，其中mask参数（F=27），以及最大时间掩码比率（ps=0.05）的十倍时间掩码，其中时间掩码的最大大小被设置为话语长度的ps倍。

## 3.2.    Conformer Transducer

通过对网络深度、模型维度、注意头数量的不同组合进行扫描，并在模型参数大小约束下选择性能最好的模型，我们分别设置了三个模型，即小模型、中模型和大模型，其参数分别为10M、30M和118M。我们在所有模型中都使用一个LSTM层decoder。表1描述了它们的体系结构超参数。

对于正则化，我们在每个confermer残差单元（即每个模块的输出）中应用dropout[29]，然后再将其添加到模块输入中。我们使用的比率为Pdorp=0.1。将多种噪声[5，30]作为正则化引入模型。网络中的所有可训练权重中还添加了一个具有1e−6权重的L2正则化。我们使用Adam优化器[31]对模型进行训练，*β*1 =0.9，*β*2 =0.98 ……以及transfomer学习率表[6]；采用10k预热步骤和峰值学习率0.05/√d，其中为d为conformer编码器中的模型尺寸。

我们使用3层LSTM语言模型（LM），维度4096，在LibriSpeech langauge模型语料库上训练，添加LibriSpeech 960h转录文本，用LibriSpeech 960h构建的1k WPM token。LM在开发集转录本上具有单词级的困惑度为63.9。浅层融合的LM权重*λ*通过网格搜索在dev集上进行调整。所有模型都是用Lingvo工具箱实现的[32]。

## 3.3.    LibriSpeech结果

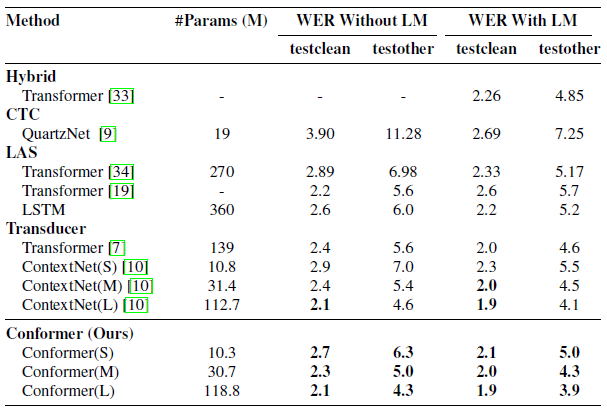
表2中我们的LibriSpeech test clean/test other模型的（WER）结果与一些最先进的模型进行了比较，这些模型包括：ContextNet[10]、Transformer transducer[7]和QuartzNet[9]。我们所有的评估结果在四舍五入到小数点后1位。

在没有语言模型的情况下，我们的中等模型在test/testother上的性能已经达到了2.3/5.0的结果，优于最著名的Transformer、基于LSTM的模型或类似规模的卷积模型。添加语言模型后，我们的模型得到了所有现有模型中最低的词错误率。这清楚地证明了在单个神经网络中结合transformer和卷积的有效性。

表1：通过扫描不同的组合并在参数范围内选择性能最好的模型，得到了符合S、M和L模型的模型超参数。

|  |  |  |  |
| --- | --- | --- | --- |
| 模型 | Conformer(S) | Conformer(M) | Conformer(L) |
| Num Params(M) | 10.3 | 30.7 | 118.8 |
| Encoder Layers | 16 | 16 | 17 |
| Encoder Dim | 144 | 256 | 512 |
| Attention Heads | 4 | 4 | 8 |
| Conv Kernel Size | 32 | 32 | 32 |
| Decoder Layers | 1 | 1 | 1 |
| Decoder Dim | 320 | 640 | 640 |

表2：conformer与最近公布的模型的比较。我们的模型在不同的模型参数大小约束下显示出一致的改进。在10.3M参数下，我们的模型在testother上比当前论文ContextNet（S）[10]好0.7%。在30.7M的模型参数下，我们的模型已经显著优于先前公布的transformer-transducer（139M参数）的最新结果[7]。



## 3.4.    对比实验（Ablation Studies）

*3.4.1. Conformer Block vs Transformer Block*

Conformer在许多方面不同于Transformer，特别是，包括卷积块并且在卷积块周围具有一对Macaron样式的ffn。下面我们研究这些差异的影响，方法是在保持参数总数不变的情况下，将conformer块变为transformer块。表3显示了每个变更对conformer块的影响。在所有差异中，卷积子块是最重要的特征，而具有Macaron风格的FFN对也比具有相同参数数目的单个FFN更有效。使用swish激活可以加快conformer模型的收敛速度。

表3：分解conformer。（1）用ReLU替换SWISH；（2）移除卷积子块；（3）用单个FFN替换Macaron风格的FFN对；（4）用普通self-attention层替换相对位置嵌入[20]的self-attention[6]。所有对比研究结果均在没有外部LM的情况下进行评估。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型结构 | dev clean | dev other | test clean | test other |
| Conformer Model | 1.9 | 4.4 | 2.1 | 4.3 |
| - SWISH + ReLU | 1.9 | 4.4 | 2.0 | 4.5 |
| - Convolution Block | 2.1 | 4.8 | 2.1 | 4.9 |
| - Macaron FFN | 2.1 | 5.1 | 2.1 | 5.0 |
| - Relative Pos. Emb. | 2.3 | 5.8 | 2.4 | 5.6 |

*3.4.2. 卷积和transformer模块的组合*

我们研究了多种不同的方法结合self-attention（MHSA）模块和卷积模块的效果。首先，我们尝试将卷积模块中的深度卷积替换为轻量级卷积[35]，发现性能显著下降，尤其是在devother数据集上。其次，我们研究了在我们的conformer模型中，将卷积模块放在MHSA模块之前，发现它会使结果降低0.1。该体系结构的另一种可能的方法是将输入分解为多头self-attention模块和卷积模块的并行分支，其输出串联在一起，如[17]所示。我们发现，这与我们提出的架构相比，性能降低。

表4中的这些结果表明了将卷积模块放在conformer块中self-attention模块之后的优势。

表4：confomer注意卷积块的对比研究。改变卷积块与多头self-attention的组合：（1）Conformer体系结构；（2）在Conformer的卷积块中使用轻量级卷积而不是深度卷积；（3）卷积位于多头self-attention前面；（4）卷积和MHSA并行，输出串联[17]。

|  |  |  |
| --- | --- | --- |
| 模型结构 | dev clean | dev other |
| Conformer | 1.9 | 4.4 |
| -Depthwise conv + Lightweight convolution | 2.0 | 4.8 |
| Convolution block before MHSA | 1.9 | 4.5 |
| Parallel MHSA and Convolution | 2.0 | 4.9 |

*3.4.3. Macaron前馈模块*

与在注意力模块后接一个前馈网络的transformer不同，conformer在自注意力和卷积模块前后，有一对macaron样式的前馈网络。此外，conformer前馈模块与半步残差一起使用。表5显示了更改conformer块以使用单个FFN或全步残差的影响。

表5：Macaron网络前馈模块的对比研究。消除了transformer模型中的conformer前馈模件与单一FFN模件之间的差异：（1）conformer前馈模件；（2）前馈模件中带有全阶跃残差的conformer前馈模件；（3）用单一FFN代替Macaron型FFN模件。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型结构 | dev clean | dev other | test clean | test other |
| Conformer | 1.9 | 4.4 | 2.1 | 4.3 |
| Single FFN | 1.9 | 4.5 | 2.1 | 4.5 |
| Full step residuals | 1.9 | 4.5 | 2.1 | 4.5 |

*3.4.4. attention head数量*

在self-attention中，每一个注意头都学会了关注输入的不同部分，这使得在简单的加权平均值之外改进预测成为可能。我们进行实验，研究在大模型中，在所有层中使用相同数量的注意头，将注意头的数量从4个改变到32个的效果。我们发现，将注意力头数量增加到16提高了准确性，特别是在devother数据集上，如表6所示。

表6：多头self-attention中注意头的对比研究。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attention数量 | 维度 | dev clean | dev other | test clean | test other |
| 4 | 128 | 1.9 | 4.6 | 2.0 | 4.5 |
| 8 | 64 | 1.9 | 4.4 | 2.1 | 4.3 |
| 16 | 32 | 2.0 | 4.3 | 2.2 | 4.4 |
| 32 | 16 | 1.9 | 4.4 | 2.1 | 4.5 |

*3.4.5. 卷积核大小*

为了研究核大小对深度卷积的影响，我们使用大模型，以{3,7,17,32,65}这些数值遍历核大小。所有层使用相同的核大小。我们发现，在内核大小为17和32之前，随着内核大小的增加，性能会提高；但在核大小为65时，性能会恶化，如表7所示。通过比较dev-WER中的第二位小数，我们发现内核大小32比其他值性能更好。

表7：深度卷积核尺寸的对比研究。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 核大小 | dev clean | dev other | test clean | test other |
| 3 | 1.88 | 4.41 | 1.99 | 4.39 |
| 7 | 1.88 | 4.30 | 2.02 | 4.44 |
| 17 | 1.87 | 4.31 | 2.04 | 4.38 |
| 32 | 1.83 | 4.30 | 2.03 | 4.29 |
| 65 | 1.89 | 4.47 | 1.98 | 4.46 |

# 4.     结论

在本文中，我们介绍了Conformer，一种集成CNN和transformer组件的体系结构，用于端到端语音识别。我们研究了每个组件的重要性，并证明了卷积模块的加入对conformer模型的性能至关重要。该模型比以往在LibriSpeech数据集上的工作具有更好的精度和更少的参数，并且在test/testother中达到了1.9%/3.9%的最新性能。

# 5.     参考文献

[1] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J.Weiss, K. Rao, E. Gonina et al., “Stateof-the-art speech recognition with sequence-to-sequence models,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 4774–4778.

[2] K. Rao, H. Sak, and R. Prabhavalkar, “Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer,” in 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2017, pp. 193–199.

[3] Y. He, T. N. Sainath, R. Prabhavalkar, I. McGraw, R. Alvarez, D. Zhao, D. Rybach, A. Kannan, Y. Wu, R. Pang, Q. Liang, D. Bhatia, Y. Shangguan, B. Li, G. Pundak, K. C. Sim, T. Bagby, S.-Y. Chang, K. Rao, and A. Gruenstein, “Streaming End-to-end Speech Recognition For Mobile Devices,” in Proc. ICASSP, 2019.

[4] T. N. Sainath, Y. He, B. Li, A. Narayanan, R. Pang, A. Bruguier, S.-y. Chang, W. Li, R. Alvarez, Z. Chen, and et al., “A streaming on-device end-to-end model surpassing server-side conventional model quality and latency,” in ICASSP, 2020.

[5] A. Graves, “Sequence transduction with recurrent neural networks,” arXiv preprint arXiv:1211.3711, 2012.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2017.

[7] Q. Zhang, H. Lu, H. Sak, A. Tripathi, E. McDermott, S. Koo, and S. Kumar, “Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss,” in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 7829–7833.

[8] J. Li, V. Lavrukhin, B. Ginsburg, R. Leary, O. Kuchaiev, J. M. Cohen, H. Nguyen, and R. T. Gadde, “Jasper: An end-to-end convolutional neural acoustic model,” arXiv preprint arXiv:1904.03288, 2019.

[9] S. Kriman, S. Beliaev, B. Ginsburg, J. Huang, O. Kuchaiev, V. Lavrukhin, R. Leary, J. Li, and Y. Zhang, “Quartznet: Deep automatic speech recognition with 1d time-channel separable convolutions,” arXiv preprint arXiv:1910.10261, 2019.

[10] W. Han, Z. Zhang, Y. Zhang, J. Yu, C.-C. Chiu, J. Qin, A. Gulati, R. Pang, and Y.Wu, “Contextnet: Improving convolutional neural networks for automatic speech recognition with global context,” arXiv preprint arXiv:2005.03191, 2020.

[11] T. N. Sainath, A.-r. Mohamed, B. Kingsbury, and B. Ramabhadran, “Deep convolutional neural networks for lvcsr,” in 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013, pp. 8614–8618.

[12] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, “Convolutional neural networks for speech recognition,” IEEE/ACM Transactions on audio, speech, and language processing, vol. 22, no. 10, pp. 1533–1545, 2014.

[13] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.

[14] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, “Attention augmented convolutional networks,” in Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3286– 3295.

[15] B. Yang, L. Wang, D. Wong, L. S. Chao, and Z. Tu, “Convolutional self-attention networks,” arXiv preprint arXiv:1904.03107, 2019.

[16] A.W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le, “Qanet: Combining local convolution with global self-attention for reading comprehension,” arXiv preprint arXiv:1804.09541, 2018.

[17] Z. Wu, Z. Liu, J. Lin, Y. Lin, and S. Han, “Lite transformer with long-short range attention,” arXiv preprint arXiv:2004.11886, 2020.

[18] Y. Lu, Z. Li, D. He, Z. Sun, B. Dong, T. Qin, L. Wang, and T.-Y. Liu, “Understanding and improving transformer from a multi-particle dynamic system point of view,” arXiv preprint arXiv:1906.02762, 2019.

[19] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang et al., “A comparative study on transformer vs rnn in speech applications,” arXiv preprint arXiv:1909.06317, 2019.

[20] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, “Transformer-xl: Attentive language models beyond a fixed-length context,” 2019.

[21] Q.Wang, B. Li, T. Xiao, J. Zhu, C. Li, D. F.Wong, and L. S. Chao, “Learning deep transformer models for machine translation,” in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Jul. 2019, pp. 1810–1822.

[22] T. Q. Nguyen and J. Salazar, “Transformers without tears: Improving the normalization of self-attention,” arXiv preprint arXiv:1910.05895, 2019.

[23] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, “Language modeling with gated convolutional networks,” in Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017, pp. 933–941.

[24] L. Dong, S. Xu, and B. Xu, “Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5884–5888.

[25] P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” arXiv preprint arXiv:1710.05941, 2017.

[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: an asr corpus based on public domain audio books,” in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015, pp. 5206–5210.

[27] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, “Specaugment: A simple data augmentation method for automatic speech recognition,” arXiv preprint arXiv:1904.08779, 2019.

[28] D. S. Park, Y. Zhang, C.-C. Chiu, Y. Chen, B. Li, W. Chan, Q. V. Le, and Y. Wu, “Specaugment on large scale datasets,” arXiv preprint arXiv:1912.05533, 2019.

[29] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” Journal of Machine Learning Research, vol. 15, no. 56, pp. 1929–1958, 2014.

[30] K.-C. Jim, C. L. Giles, and B. G. Horne, “An analysis of noise in recurrent neural networks: convergence and generalization,” IEEE Transactions on neural networks, vol. 7, no. 6, pp. 1424– 1438, 1996.

[31] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” arXiv preprint arXiv:1412.6980, 2014.

[32] J. Shen, P. Nguyen, Y. Wu, Z. Chen, and et al., “Lingvo: a modular and scalable framework for sequence-to-sequence modeling,” 2019.

[33] Y. Wang, A. Mohamed, D. Le, C. Liu, A. Xiao, J. Mahadeokar, H. Huang, A. Tjandra, X. Zhang, F. Zhang et al., “Transformerbased acoustic modeling for hybrid speech recognition,” arXiv preprint arXiv:1910.09799, 2019.

[34] G. Synnaeve, Q. Xu, J. Kahn, T. Likhomanenko, E. Grave, V. Pratap, A. Sriram, V. Liptchinsky, and R. Collobert, “End-toend asr: from supervised to semi-supervised learning with modern architectures,” 2019.

[35] F. Wu, A. Fan, A. Baevski, Y. N. Dauphin, and M. Auli, “Pay less attention with lightweight and dynamic convolutions,” arXiv preprint arXiv:1901.10430, 2019.

1. *ContextNet: Improving Convolutional Neural Networks for Automatic Speech Recognition with Global Context*：使用squeeze-and-excitation模块提取全局信息加入到卷积层，提高ASR效果。2020. [↑](#footnote-ref-1)
2. *Attention augmented convolutional networks*：将CNN和self-attention特征组合起来，用于图像分类/目标检测。2019. [↑](#footnote-ref-2)
3. *Lite transformer with long-short range attention*：集合attention和cnn的用于手机端的轻量级模型，解决nlp任务。2020. [↑](#footnote-ref-3)
4. *Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss*：rnn-t结构中的rnn部分替换为transformer，限制self-attention的上下文使支持流式解码。2020. [↑](#footnote-ref-4)
5. *Transformer-xl: Attentive language models beyond a fixed-length context*：2019. [↑](#footnote-ref-5)
6. *Understanding and improving transformer from a multi-particle dynamic system point of view*：2019. [↑](#footnote-ref-6)