Streaming Chunk-Aware Multihead Attention for Online End-to-End Speech Recognition

在线端到端语音识别中的流块感知多头注意

*张士良*1*，高芝芙*1*，罗浩能*2*，明磊*1*，高洁*1*，颜志杰*1*、谢磊*2 1

阿里巴巴达摩学院语音实验室2

ASLP@NPU公司，西北工业大学，中国

{斯莱.zsl, 芝罘.gzf}@阿里巴巴-公司网站, 邮箱：lxie@nwpu.edu.cn

# 摘要

Recently, streaming end-to-end automatic speech recognition (E2E-ASR) has gained more and more attention. Many efforts have been paid to turn the non-streaming attention-based E2E-ASR system into streaming architecture. In this work, we propose a novel online E2E-ASR system by using Streaming Chunk-Aware Multihead Attention (SCAMA) and a latency control memory equipped self-attention network (LC-SAN-M). LC-SAN-M uses chunk-level input to control the latency of encoder. As to SCAMA, a jointly trained predictor is used to control the output of encoder when feeding to decoder, which enables decoder to generate output in streaming manner. Experimental results on the open 170-hour AISHELL-1 and an industrial-level 20000-hour Mandarin speech recognition tasks show that our approach can significantly outperform the MoChA-based baseline system under comparable setup. On the AISHELL-1 task, our proposed method achieves a character error rate (CER) of 7.39%, to the best of our knowledge, which is the best published performance for online ASR.

Index Terms: Automatic Speech Recognition, End-to-End, SCAMA, online ASR, LC-SAN-M

近年来，流式端到端自动语音识别（E2E-ASR）得到了越来越多的关注。将基于非流式注意力的E2E-ASR系统转化为流式体系结构是目前研究的热点。在这项工作中，我们提出了一个新的在线E2E-ASR系统，使用流式区块感知多头注意（SCAMA）和一个延迟控制记忆的自我注意网络（LC-SAN-M）。LC-SAN-M使用块级输入来控制编码器的延迟。在SCAMA算法中，采用联合训练的预测器对编码器的输出进行控制，使解码器能够以流的方式产生输出。在开放的170小时AISHELL-1和工业级20000小时普通话语音识别任务上的实验结果表明，在可比较的设置下，我们的方法可以显著地优于基于MoChA的基线系统。在AISHELL-1任务上，我们提出的方法获得了7.39%的字符错误率（CER），这是在线ASR的最佳性能。

索引词：自动语音识别，端到端，SCAMA，在线ASR，LC-SAN-M

# 1.     介绍

端到端（E2E）自动语音识别（ASR）越来越受到语音识别界的重视。与传统的混合ASR系统不同，E2E系统将声学模型（AM）、语言模型（LM）和语音模型（PM）合并为一个序列到序列模型，大大简化了训练和解码流程。目前，有三种流行的端到端方法，即连接主义时间分类（CTC）[1]、递归神经网络变换器（RNN-T）[2]和基于注意的编解码器（AED）[3,4,5]。CTC提出了一个独立性假设，即标签输出在条件上相互独立。因此，它通常需要与外部语言模型相结合才能获得良好的识别效果[6，7]。与基于CTC的模型不同，RNN-T和基于注意的编解码器（AED）模型没有独立性假设，即使没有外部语言模型，也能获得最先进的性能。

最具代表性的基于注意的模型是所谓的LAS[5]，它由一个基于金字塔双向长短时记忆（BLSTM）的编码器和一个基于注意的LSTM解码器组成。编码器将原始的声学特征转化为更高层次的表示，而具有注意机制的解码器基于先前的预测以自回归的方式预测下一个输出符号。解码器中的注意模块用于计算动态软对齐并生成上下文向量。正如最初定义的，软注意需要在每个输出时间步关注整个输入序列。因此，基于软注意的E2E模型不适用于在线语音识别，因为它必须等到输入序列被处理后才能产生输出。

在以往的工作中，人们一直致力于将全序列的软注意转化为局部注意，这是一种适合于在线语音识别的方法。在[8]中，基于输入和输出序列元素之间的对齐本质上是单调的，提出了一种硬单调注意。沿着这条思路，文献[9]提出了一种改进的单调注意，即单调分块注意（MoChA），它使得该模型能够在硬单调注意机制选择参与的小块记忆上进行软注意。实验结果表明，MoChA[9]及其变体AMoChA[10]和sMoChA[11]有效地缩小了语音识别任务中单调注意和软注意之间的差距。同时，在[12，13，14]中的工作通过仅在局部窗口上计算能量值来利用局部注意。MoChA类型和基于局部窗口的注意都使用一些预先设置的超参数来截断输入序列以实现在线注意，例如MoChA中用于停止扫描内存的阈值和基于局部窗口的注意中的窗口大小。在实际应用中，这些预设的超参数可能会使这些在线关注不够健壮。最近，流E2E-ASR的研究者们尝试将注意力和额外的对齐信息结合起来来执行流截断。在[15]中，提出的触发注意（TA）使用基于CTC的分类器来动态控制基于注意的解码器神经网络的激活。在文献[16]中，提出了一种用词级forcealignment训练的Scout网络来流式检测词边界，而不需要看到任何未来的帧。

编码器架构是流E2E-ASR的另一个关键元素。为了控制延迟，以前的工作通常采用单向LSTM[9]或延迟控制双向LSTM（LC-BLSTM）[10,11]。最近，基于自我关注的Transformer[17]在E2EASR[18，19]中变得流行起来。改进的关键是在编码器和译码器中使用了自关注机制而不是递归机制，从而提高了捕获长距离依赖的能力，同时降低了计算复杂度，提高了训练的并行性。在[20，21]中，变压器被进一步设计成能够执行在线编码的结构。

在这项工作中，我们提出了一个新的基于流式块感知的多头部注意（SCAMA）的在线E2E-ASR系统。对于编码器，我们将我们以前的配备内存的自注意（SAN-M）[22]扩展到延迟控制ar-

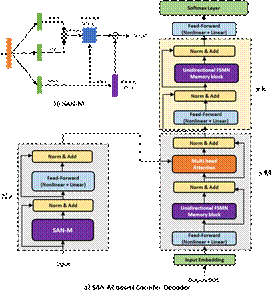


图1：配备存储器的基于自我注意的编码器-解码器的图示。

体系结构，即LC-SAN-M。基于LC-SAN-M的编码器使用块级输入来控制编码器延迟。对于SCAMA，我们使用一个联合训练的预测器来预测每个块中的令牌数量，并控制基于注意的解码器的激活。与触发注意（TA）相比，预测因子的训练采用交叉熵损失代替CTC损失。更重要的是，预测块级输入中的令牌数可以达到很高的精度，从而消除训练和测试之间的不匹配。我们已经评估了我们的方法对公众170小时AISHELL-1和工业级20000小时普通话识别任务。与原始的全序列注意相比，基于SCAMA的在线E2E-ASR系统的性能下降可以接受，并且比基于MoChA的基线系统获得了更好的性能。

# 2.        E2E-ASR，记忆装备自我注意

在我们之前的工作[22]中，我们提出了一种在编解码器框架下用于端到端语音识别的带记忆的自注意（SAN-M）。在本节中，我们将对基于SAN-M的模型进行简要的回顾。如图1 a）所示，编码器由SAN-M和前馈组件块组成，解码器由配备有多头注意力的单向深度前馈顺序存储网络（DFSMN）[24,25]层块和纯单向DFSMN层块组成。至于SAN-M，它将Transformer中的多头自注意[17]与FSMN中的内存块结合起来，如图1b所示。这种与上下文无关的FSMN记忆块和上下文相关的自我注意的结合产生了强大的局部和长期依赖建模能力。给定一个输入序列，表示为={x1，···xt，···xt}T，其中每个∈rre表示时间实例的输入数据。SAN-M的具体操作如下。*不米K公司***十***文本d级*×1*t型*

                     多头（X）=[head1，…，headh]WO（1）

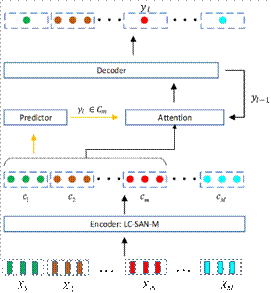


图2:SCAMA在线E2E-ASR系统示意图。

headi=自我攻击（Qi，Ki，Vi）=软最大值(

                           （Qi，Ki，Vi）=（XWiQ，XWiK，XW）(3)

其中投影是参数矩阵∈R×d，∈R×d，∈R×d和∈R×hd。是头数，是模型尺寸。在这项工作中，多头注意力由4个头部组成（=4）。FSMN存储器块的输出可计算如下。**世界大战***我问d级模型问iK公司d级模型k公司四d级模型五Od级模型五小时d级模型小时*

(4)

**米**=[m1，m2，··，mT]吨(5)

这里，表示自我注意值中的第次实例。和分别是FSMN内存块的向后和向前顺序。表示两个大小相等的向量按元素相乘。*及物动词t型我`左后*

**是的**=多头（XM）+(6)

**是的**表示SAN-M的输出。其他操作前馈、规范和多磁头注意与原始变压器相同[17]，单向FSMN与原始FSMN相同[24]。*添加*

# 3.     在线E2E-ASR

所提出的在线端到端语音识别系统的总体架构如图3所示。与原来的基于SAN-M的E2E模型相比，有两个方面的变化使得系统可以流化。在编码器方面，我们将SAN-M扩展到一个延迟控制版本，即LC-SAN-M。在注意模块方面，我们提出了一种新的流块感知多头注意。

## 3.1.    LC-SAN-M公司

对于文献[22]中原有的基于记忆的自注意（SAN-M）编码器，全序列注意机制使其不适用于在线ASR。为了控制编码器延迟，我们将SAN-M扩展到LC-SANM。输入序列根据预设的块大小被划分为块级，表示为=**X X X***c级*

{[x1，···，xc]，[xc+1，···，x2c]，···，[xnc+1···xT]}T。块大小与编码器延迟有关。在这项工作中，我们将评估值为5、10和10的各种块大小*c级*

15在符号上，={[xkc+1，···，x（k+1）c]}T表示第个块中的样本。对于第个区块中的每个时间实例，它只能访问当前区块和以前区块中的样本。因此，可以使用以下公式来计算LC-SAN-M的输出。**XX号***k公司k公司k公司k公司*



|  |  |
| --- | --- |
| **K公司**¯*我*（k） =[k（k−1）；Ki（k）]¯*我* | (8) |
| **五**¯*我*（k） =[V（k−1）；Vi（k）]¯*我* | (9) |
| 头i（k）=自攻（Qi（k），k（k），V（k））¯*我*¯*我* | (10) |
| 多头（Xk）=[头1（k），…，头H（k）]WO | (11) |

此外，将等式（4）修改为以下单向FSMN存储器块。



**米**(13)

这里是FSMN内存块的总过滤顺序。最后，我们可以得到如下LC-SAN-M的输出。*我***十***k公司*

**是的***k公司*=多头（XMk）+k(14)

## 3.2.    斯卡玛

如图3所示，我们在编码器的顶部堆叠了一个预测器，该预测器经过训练可以预测每个块中的令牌数。编码器的分块输出被拼接，然后输入预测器。值得注意的是，让我们将第-个块的拼接输出表示为。然后预测器生成概率为：*k公司***小时***s码k公司主键*

|  |  |
| --- | --- |
| *主键*=softmax（最大值（hskW1+b1,0）W2+b2）  使用交叉熵损失对预测值进行训练： | (15) |
| L=−Xt（p）*pred公司k公司日志k公司* | (16) |

*k公司*

并利用以下损失函数对整个系统进行联合优化：

                                      L=L2e+αL（17）*电子pred公司*

这里，表示第块中的地面真值令牌数的一个热向量，为0.2。Le2e是训练编解码器的原始CE损耗。我们使用训练有素的基于CTC的ASR系统[23]来生成帧级对齐，然后将它们转换为块级标签。我们首先计算训练集中块中包含的最大令牌数。因此，预测器的训练被描述为一个多类分类问题。在训练过程中，地面真值令牌号被用来引导编码器输出馈入解码器。如果解码器的第个令牌在第个块中，那么表1:AISHELL-1上各种E2E模型的性能。（FSA表示全序列注意。）*塔卡k公司α`米*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 经验 | 编码器 | 解码器 | 敬告 | CER（%） |
| 1 | 圣M（10） | DFSMN（三） | 金融服务管理局 | 6.46 |
| 2  3 | LC-SAN-M（10个） | DFSMN（三） | 金融服务管理局 | 6.92 |
| 斯卡玛 | 7.39 |
| 4  5 | LC-SAN-M（10个） | LSTM（3） | 金融服务管理局 | 8.78 |
| 摩卡 | 9.01 |

表2:AISHELL-1任务系统比较。“Y”表示“是”，“N”表示“否”。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型 | E2E型 | 流光溢彩 | 在线的 | CER（%） |
| 拉斯[27] | 是的 | 是的 | 不 | 8.71 |
| 反恐委员会和注意事项[28] | 是的 | 是的 | 不 | 6.70 |
| TDNN-LFMMI[29] | 不 | 是的 | 是的 | 7.62 |
| 变形金刚摩卡[20] | 是的 | 是的 | 是的 | 9.7 |
| LC-SAN-M-SCAMA公司 | 是的 | 不 | 是的 | 7.39 |

只对块输入注意模块，为解码器生成上下文向量。在推理过程中，选择概率最大的类作为预测器的输出，用来指导译码器对当前输入块的处理步骤。*c级*1 *厘米*

## 3.3.    解码策略

对于基于编码-解码器的E2E-ASR，当预测到句末（eos）标记时，推理终止。对于流式E2E-ASR，其中一个问题是解码器可能过早或过晚地预测EOS令牌[26]。在我们的工作中，我们还发现解码器可能过早地生成eos令牌，特别是当块大小很小时。我们提出了一个技巧，在波束搜索解码的基础上处理这个问题。在推理过程中，如果解码器生成一个eos令牌，并且输入的不是最后一个块，那么我们将使用前一个令牌和历史信息来预测下一个令牌，而不是eos。对于最后一个块，如果预测的令牌数为，则总解码步骤将为1到+2。当解码器在最后一个块中生成eostoken或解码+2步时，推断终止。*<><><><><>不不<>不*

# 4.     实验

我们已经评估了我们提出的基于SCAMA的在线E2E-

两个普通话语音识别任务的ASR，即[29]中发布的AISHELL-1任务和一个20000小时的普通话任务。对于AISHELL-1任务，我们使用150小时的训练集进行模型训练，使用10小时的开发集进行早期停止。最后，在7176个句子（约5小时）的测试集中报告了字符错误率（CER%）。对于2万小时的普通话任务，与文献[23]相同，它包括从新闻、体育、旅游、游戏、文学、教育等多领域收集的约2万小时的数据，按照95%-5%的比例分为训练集和发展集。远场数据集由约15小时的数据组成，而公共数据集由约30小时的数据组成，用于评估性能。

用于所有实验的声学特征是80维log-mel滤波器组（FBK）能量，在25ms窗口上以10ms偏移计算。我们在7（3+1+3）的上下文窗口中堆叠连续的帧以产生三维特征，然后对输入帧进行下采样

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 表3:20000小时任务中各种E2E模型的性能。   |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 模型类型 | 反恐委员会 | | 不 | 关于流媒体E2E | |  | 流式E2E | | | | | 型号ID | CTC1型 | CTC2型 | E2E1型 | E2和E2 | E2E3型 | E2E4型 | E2E5型 | E2E6型 | E2E7型 | E2E8型 | | 编码器 | DFSMN（10） | DFSMN（20个） | 圣M（40） |  | | LC-SAN-M（40个） | | | | | | 解码器 | - | - | DFSMN（12） | DFSMN（12） | | LSTM（4） | | DFSMN（12） | | | | 敬告 | - | - | 金融服务管理局 | 金融服务管理局 | 金融服务管理局 | 金融服务管理局 | 摩卡 | 斯卡玛 | | | | 编码器延迟 | 600毫秒 | 1.2秒 | 完全 | 600毫秒 | 900毫秒 | 600毫秒 | 600毫秒 | 300毫秒 | 600毫秒 | 900毫秒 | | 总延迟 | 600毫秒 | 1.2秒 | 完全 | 完全 | 完全 | 完全 | 600毫秒 | 300毫秒 | 600毫秒 | 900毫秒 | | 参数（M） | 25 | 45 | 42 | 42 | 42 | 60 | 60 | 43 | 43 | 43 | | 公共集（CER%） | 11.6 | 9.9 | 9.0 | 9.9 | 9.4 | 10.1 | 11.5 | 11.4 | 10.4 | 9.8 | | 远场集（CER%） | 20.3 | 17.7 | 13.7 | 15.1 | 14.9 | 15.8 | 18.1 | 17.0 | 16.0 | 15.2 | |

速率为60ms。声学建模单位为汉字，AISHELL-1和20000hour任务分别为4233和9000。我们使用Tensorflow[30]以分布式方式训练模型。添加了标签平滑和值为0.1的脱落正则化，以防止过拟合。SpecAugment[31]也用于所有实验。

## 4.1.    AISHELL-1任务

在表1中，我们评估了各种E2EASR系统在AISHELL-1任务上的性能。对于基线模型，我们训练了基于SAN-M的E2E-ASR系统[22]。如图1所示，我们设置=10、=3和=0，这意味着编码器由10个SAN-M块组成，解码器由三个配备全序列注意（FSA）的DFSMN层组成。编码器和解码器的线性和非线性层尺寸分别为512和2048。该模型的CER为6.46%。然后，我们将基于SAN-M的编码器替换为基于LC-SAN-M的编码器，以研究编码器延迟对性能的影响。LCSAN-M的块大小是10。表1中exp1和exp2的比较表明，它的相对性能下降了约7%。此外，我们将全序列注意（FSA）替换为所提出的SCAMA。如表1的exp3所示，它的相对性能下降了6.8%。为了评估解码器架构的影响并与MoChA进行比较，我们进行了exp4和exp5。两个解码器都由三个LSTM层和512个单元组成。实验结果表明，用基于MoChA的在线注意代替全序列注意，在这项任务中会出现较小的性能下降。然而，基于LSTM解码器的系统的性能远远落后于基于DFSMN解码器的系统。在表2中，我们将我们提出的系统与其他已发布的系统进行了比较。我们提出的LCSAN-M和基于SCAMA的在线E2E-ASR系统在不使用任何外部LM的情况下实现了7.39%的CER。据我们所知，这是最先进的性能在线ASR系统在这项任务。*不米K公司*

## 4.2.    20000小时任务

在本任务中，我们比较了三种类型的E2E-ASR系统：CTC、非流式E2E和流式E2E。对于基于CTC的系统，如[23]中所述，我们训练了两个DFSMN-CTCsMBR系统，分别具有10层和20层DFSMN，在表3中分别表示为CTC1和CTC2。基于CTC的模型用一个外部的5-gram语言模型进行解码。对于非流E2E-ASR系统，我们训练了四个模型，在表3中表示为E2E1到E2E4。对于E2E1系统，我们设置了=40、=6和=6，这意味着解码器由12层DFSMN组成，底部6层配备了全序列多头注意。线性和非线性层的大小分别为256和1024。对于E2E2到E2E4，我们*不米K公司*

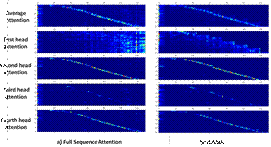


图3:FSA和SCAMA的图示。x轴是指声音帧，y轴是指字符。

将基于序列级SAN-M的编码器替换为基于LCSAN-M的编码器。编码器块大小对于E2E2和E2E4是10，对于E2E3是15。由于我们使用6对输入进行下采样，因此相应的编码器延迟分别为600ms和900ms。对于E2E4，我们进一步用LSTM替换解码器，LSTM由4个LSTM层和768个单元组成。对于流式E2E-ASR系统，我们评估了基于MoChA的模型（E2E5）和基于SCAMA的模型，其块大小为5、10和15（E2E6到E2E8）。当使用32个NVIDIA特斯拉V100 GPU时，大约需要3天时间来训练SCAMA模型，比摩卡系统快3倍。

在表3中，我们总结了各种系统在20000小时任务中的性能。对于非流E2E-ASR系统，当编码器的未来上下文信息有限时，性能将降低。与E2E2和E2E4相比，基于DFSMN译码器的系统比基于LSTM译码器的系统具有更好的性能和更小的模型尺寸。对于E2E4和E2E5来说，当全序列注意力被在线摩卡注意力取代时，性能将受到显著损失。这个实验现象与第二节中的AISHELL-1任务不同。4.1. 根据我们的实验分析，这是由于MoChA在推理过程中使用预设的阈值来停止对记忆的扫描，对噪声语音不具有鲁棒性。将基于SCAMA的系统（E2E6到E2E8）与非流式系统（E2E1到E2E3）进行比较，结果表明基于SCAMA的在线注意存在可接受的性能下降。图3是E2E1和E2E7最后一层的注意力可视化。全序列多头注意的总体趋势是单调的。然而，似乎编码器和解码器之间的注意不仅起到了对齐的作用，而且还进行了上下文建模。因此，将注意力限制在局部窗口或完全单调会导致性能下降。对于基于SCAMA的系统来说，性能退化比基于MoChA的系统要小，因为它只会限制未来的信息。

# 5.     结论

本文提出了一种新颖的在线端到端语音识别系统。特别地，我们提出了一个基于流块感知的多头注意（SCAMA）和一个基于延迟控制记忆的自注意（LC-SAN-M）的在线E2E-ASR系统。与全序列注意相比，SCAMA的性能下降是可以接受的。在AISHELL-1任务中，我们提出的在线E2E-ASR系统在不使用任何外部LM的情况下实现了7.39%的CER。在一个20000小时的普通话任务中，基于SCAMA的在线E2EASR系统可以显著优于基于CTC和MoChA的具有相同延迟的系统。

# 6.     工具书类

[1] A. Graves, S. Fern´andez, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks,” in Proceedings of the 23rd international conference on Machine learning, 2006, pp. 369–376.

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

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

[4] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, “Attention-based models for speech recognition,” in Advances in neural information processing systems, 2015, pp. 577– 585.

[5] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016, pp. 4960–4964.

[6] A. Graves and N. Jaitly, “Towards end-to-end speech recognition with recurrent neural networks,” in International conference on machine learning, 2014, pp. 1764–1772.

[7] H. Sak, A. Senior, K. Rao, and F. Beaufays, “Fast and accurate recurrent neural network acoustic models for speech recognition,” arXiv preprint arXiv:1507.06947, 2015.

[8] C. Raffel, M.-T. Luong, P. J. Liu, R. J.Weiss, and D. Eck, “Online and linear-time attention by enforcing monotonic alignments,” in Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017, pp. 2837–2846.

[9] C.-C. Chiu and C. Raffel, “Monotonic chunkwise attention,” arXiv preprint arXiv:1712.05382, 2017.

[10] R. Fan, P. Zhou, W. Chen, J. Jia, and G. Liu, “An online attention-based model for speech recognition,” arXiv preprint arXiv:1811.05247, 2018.

[11] H. Miao, G. Cheng, P. Zhang, T. Li, and Y. Yan, “Online hybrid ctc/attention architecture for end-to-end speech recognition,” Proc. of Interspeech 2019, pp. 2623–2627, 2019.

[12] J. Hou, S. Zhang, and L.-R. Dai, “Gaussian prediction based attention for online end-to-end speech recognition.” in INTERSPEECH, 2017, pp. 3692–3696.

[13] A. Tjandra, S. Sakti, and S. Nakamura, “Local monotonic attention mechanism for end-to-end speech and language processing,” arXiv preprint arXiv:1705.08091, 2017.

[14] A. Merboldt, A. Zeyer, R. Schl¨uter, and H. Ney, “An analysis of local monotonic attention variants,” Proc. of Interspeech 2019, pp. 1398–1402, 2019.

[15] N. Moritz, T. Hori, and J. Le Roux, “Triggered attention for endto- end speech recognition,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 5666–5670.

[16] C.Wang, Y.Wu, S. Liu, J. Li, L. Lu, G. Ye, and M. Zhou, “Reducing the latency of end-to-end streaming speech recognition models with a scout network,” arXiv preprint arXiv:2003.10369, 2020.

[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in Advances in neural information processing systems, 2017, pp. 5998–6008.

[18] 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.

[19] N.-Q. Pham, T.-S. Nguyen, J. Niehues, M. Muller, and A.Waibel, “Very deep self-attention networks for end-to-end speech recognition,” arXiv preprint arXiv:1904.13377, 2019.

[20] E. Tsunoo, Y. Kashiwagi, T. Kumakura, and S. Watanabe, “Towards online end-to-end transformer automatic speech recognition,” arXiv preprint arXiv:1910.11871, 2019.

[21] N. Moritz, T. Hori, and J. L. Roux, “Streaming automatic speech recognition with the transformer model,” arXiv preprint arXiv:2001.02674, 2020.

[22] Z. Gao, S. Zhang, and M. Lei, “SAN-M: Memory equipped selfattention for end-to-end speech recognition,” in Submitted to INTERSPEECH 2020.

[23] S. Zhang, M. Lei, Y. Liu, and W. Li, “Investigation of modeling units for mandarin speech recognition using DFSMN-CTCsMBR,” in 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 7085– 7089.

[24] S. Zhang, C. Liu, H. Jiang, S. Wei, L. Dai, and Y. Hu, “Nonrecurrent neural structure for long-term dependence,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 4, pp. 871–884, 2017.

[25] S. Zhang, M. Lei, Z. Yan, and L. Dai, “Deep-FSMN for large vocabulary continuous speech recognition,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5869–5873.

[26] B. Li, S.-y. Chang, T. N. Sainath, R. Pang, Y. He, T. Strohman, and Y. Wu, “Towards fast and accurate streaming end-to-end asr,” in ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020, pp. 6069– 6073.

[27] C. Shan, C. Weng, G. Wang, D. Su, M. Luo, D. Yu, and L. Xie, “Component fusion: Learning replaceable language model component for end-to-end speech recognition system,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 5361–5635.

[28] 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.

[29] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, “AISHELL-1: An open-source mandarin speech corpus and a speech recognition baseline,” in 2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA). IEEE, 2017, pp. 1–5.

[30] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard et al., “Tensorflow: A system for large-scale machine learning,” in 12th fUSENIXg Symposium on Operating Systems Design and Implementation (fOSDIg 16), 2016, pp. 265–283.

[31] 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.