arXiv:2304.04991v1 [cs.SD] 11 Apr 2023

Sim-T：通过复用技术简化用于语音识别的 Transformer 网络

Guangyong Wei、Zhikui Duan、Shiren Li、Guangguang Yang、Xinmei Yu、Junhua L i

摘要—**近年来，Transformer 网络凭借出色的模型性能，在语音识别任务中受到广泛关注。然而，Transformer 网络始终需要大量计算，并包含大量参数，给计算资源或存储空间有限的设备带来了严重的部署问题。本文提出了一种名为 Si m-T 的新型轻量级模型，以拓展 Transformer 模型的通用性。借助新开发的复用技术，Sim-T 能够在几乎不损失性能的情况下高效压缩模型。具体而言，所提出的技术包括两个部分，即模块权重复用和注意力分数复用。此外，本文还提出了一种新颖的解码器结构，以促进注意力分数复用。我们进行了大量实验，以验证 Sim-T 的有效性。在 AISHELL-1 数据集上，当所提出的 Sim-T 参数量比基线 Transformer 少 48% 时，字符错误率可提升 0.4%。或者，当 Sim-T 达到与基线 Transformer 相同的性能时，参数量可减少 69%。对于 HKUST 和 WSJ *eval92*数据集，当 Sim-T 的参数量比基线 Transformer 少 40% 时，字符错误率和词错误率分别可提升 0.3% 和 0.2%。**

索引词—**语音识别、Transformer、轻量化、权重复用、注意力分数复用**

I. 引言

TRANSFORMER网络已广泛应用于多个场景，例如自然语言处理（NLP） [2]–[4], 视觉任务 [5]–[7],语言模型（LM） [8], [9]和自动语音识别（ASR） [10]–[15]。特别是，Transformer及其变体基于注意力机制与前馈神经网络（FFN）的结合，在自动语音识别领域的性能优于其他网络模型。然而，基于Transformer的模型往往存在参数量大、计算开销高的问题。

庞大的参数量需要充足的存储空间和内存，这使得这些高性能模型不适用于计算资源有限的设备。因此，模型压缩成为解决上述问题的理想方案。权重共享最初由 [16] and [17], 应用于神经网络，已被证明是压缩模型的有效方法。最近， [18]将权重共享的理念应用于Transformer 网络，以用于机器翻译和语言模型领域中的模型压缩。随后，ALBert网络中提出了跨层参数共享方案 [3],，其中所有层的对应模块共享权重。如 [19], 中所述，随着层索引差异的增加，特征往往变得越来越不同。因此，相邻层的特征是相似的。在这种情况下，共享所有网络层中的对应模块的权重（例如ALBert网络 [3], ）并非最优方案。此外，在自动语音识别领域，使用权重共享来有效减少参数的作用尚未得到充分探索。为了有效解决自动语音识别任务中的模块压缩问题，提出了一种称为模块权重复用的新方案。在该方案中，首先将所有网络层划分为若干组，然后复用每组对应模块的权重。如图1 红色虚线所示，多个FFN 模块和各种注意力模块在每组或每层内执行权重复用。所提出的方案可以系统地探索划分的组数实际上如何影响模型性能和模型参数数量。

据公开文献所述，除了本文提出的模块权重复用外，尽管注意力分数复用同样有助于压缩基于Transformer的网络模型，但此前尚未有人对此进行研究。值得一提的是，在Transformer 网络中，用于表示输入词元不同位置之间全局交互的注意力分数，受到输入二次计算复杂度的严重影响，因而会占用大量计算资源。 [20] 研究了注意力矩阵的对角化程度，并注意到相邻层累积注意力对角线之间具有高度相似性。从这一角度来看，注意力分数复用有助于压缩基于Transformer的模型并降低计算开销。具体而言，对于每个组，注意力分数仅在第一个注意力模块中生成，然后在所有后续注意力模块之间共享。同时，标签序列作为第一个解码器层的输入，其上下文信息会随着模型层数加深而不断减弱。所提出的注意力分数复用可以很好地解决这一问题。然而，注意力分数复用技术无法直接应用于Transformer解码器层。为解决这一问题，我们开发了一种适用于注意力分数复用技术的新型解码器层。值得注意的是，使用数量更少的新型解码器层，也能达到与原始Transformer解码器层相同的性能

Guangyong Wei、Zhikui Duan、Xinmei Yu、Guangguang Yang和Junhua Li来自中国广东省佛山市佛山大学。（电子邮箱：gy.wei@foxmail.com；duanzhikui@outlook.com；labxmyu@fosu.edu.cn；sun- shineuop@163.com；jh.io@foxmail.com）
Shiren Li来自中国广州中山大学。（电子邮箱：lishr6@mail3.sysu.edu.cn）
通讯作者：Zhikui Duan

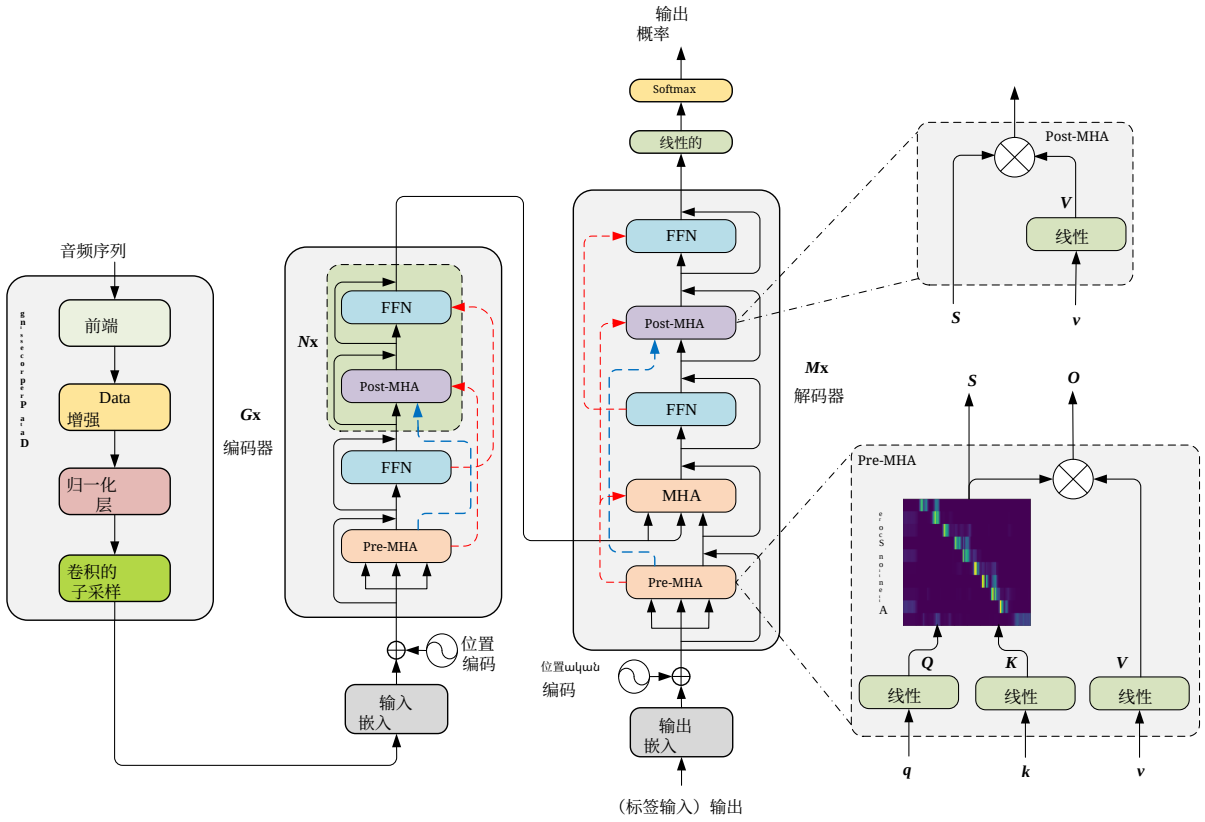

图 1. 所提出的 Sim-T 网络的整体结构，包括三个部分：数据预处理、编码器和解码器。 $G\times$ 表示所有编码层被划分为 G 组。红色虚线表示模块权重复用技术。蓝色虚线表示注意力分数复用。

。从这个意义上说，模型参数也可以减少。

在本研究中，我们提出了一种名为 Sim-T 的轻量级网络，通过复用技术简化 Speech-Transformer 网络 [10]。所提出的复用技术由两部分组成：模块权重复用和注意力分数复用。对于编码器，我们将网络层划分为多个组，并对每个组同时应用模块权重复用和注意力分数复用。模块权重和注意力分数首先由每组的第一层进行更新，然后由同组的其他层重复使用。对于解码器，我们将模块权重复用技术应用于每个网络层，并借助注意力分数复用，在每个网络层中重复使用标签的注意力分数。本文的贡献可总结如下：

- 一种名为 Sim-T 的轻量级网络被提出用于自动语音识别任务，它采用了新开发的模块权重复用和注意力分数复用技术。
- 提出了一种新颖的解码器结构，以增强标签信息并使所提出的注意力机制能够】【。

分数复用可直接应用于语音 Transformer 网络。

- Sim-T 能取得令人期待的结果。在 AISHELL-1 数据集上，当所提出的 Sim-T 比基线 Transformer 少 48 % 的参数时，字符错误率可提升 0.4%。当 Sim-T 的性能与基线 Transformer 相同时，可减少 69% 的参数。在香港科技大学数据集和 WSJ*eval92* 数据集上，当所提出的 Sim-T 比基线 Transformer 少 40% 的参数时，字符错误率和词错误率分别可提升 0.3% 和 0.2%。

II. 相关工作

近年来，自动语音识别领域对端到端网络给予了广泛关注 [21]，例如 CTC [22] 和编码器-解码器模型。尤其是编码器-解码器模型已得到广泛研究，并发表了许多相关成果，例如 RNN [23]，LSTM [24]，LAS[25]和 Transformer [1]。

Transformer 的出色模型性能主要源于自注意力机制与前馈网络的结合。Transformer 最初是为翻译任务提出的，后来受到自动语音识别领域的青睐。Speech-transformer [10], 就是一个很好的例子，是一种应用

纯 Transformer 框架。受语音Transformer 网络的启发，随后开展了大量研究[11]–[14] 。

模型压缩是一个热门研究领域。 [19] 和[26] 表明，基于深度学习的模型中存在大量冗余信息。对模型性能起决定作用的参数只占很小一部分。此外，模型压缩可以在几乎不牺牲模型性能的情况下去除冗余信息并降低计算成本。因此，轻量化模型吸引了研究人员的广泛关注。以下是一些典型案例。 [4] 采用量化感知训练方法来压缩模型规模并加快推理速度。 [27]和 [28]使用知识蒸馏方法来减少模型参数。 [29] 精简了 Transformer 网络各层的输入词元，以减少冗余信息。ALBert [3] 采用参数分解和跨层参数共享来充分压缩模型规模，从而避免参数随着网络层深度的增加而增长。

权重共享的概念最初由 [16]和 [17] 引入神经网络，并提出了一种简单有效的模型压缩方法。目前，人们已探索出一些基于权重共享的优秀方法，例如 [3], [18], [30], [31], ，以解决 Transformer 网络面临的计算量大、参数数量多的问题。具体而言，通用 Transformer [18] 在网络层中采用权重共享，从而实现了类 RNN 循环网络的效果。此外，通用 Transformer 与原始 Transformer 的不同之处在于，原始 Transformer 网络的编码器层和解码器层数量是固定的。ALBert [3] 采用跨层参数共享，避免模型规模随着网络层数的增加而扩大。通用 Transformer 和 ALBert 默认在各层之间共享所有模块参数。我们提出的 Sim-T 系统地探究了在不同网络深度应用权重共享对模型参数和模型性能的影响。考虑到 Transformer 网络中注意力分数的计算复杂度较高，我们进一步将权重复用的范围扩展到注意力分数复用，以实现模型压缩。

III. 方法

本节将介绍我们提出的 Sim-T 模型的整体结构，包括数据预处理、编码器和解码器。此外，还将阐述模块权重复用和注意力分数复用的原理。

A. Sim-T 的整体结构

端到端 Transformer [10]模型在自动语音识别领域展现出卓越性能。Transformer 模型利用多头注意力（MHA）机制，能够捕捉长距离上下文交互关系，并通过前馈网络提取高层表示。网络。所提出的 Sim-T 网络是一种通过复用技术简化的轻量级基于Transformer的网络，因此它们具有相似的优势。整体结构如图 1 所示。通过数据预处理获得的音频特征被输入编码器，从而能够很好地获取高层声学特征。编码器的输出和标签特征通过解码器进行解码。

数据预处理：在 Sim-T 模型中，首先对输入音频序列进行数据预处理，如图 1 左侧所示。原始音频序列通过前端模块从时域转换到频域，从而获得对数梅尔特征。随后，通过数据增强模块对对数梅尔频谱图进行时间扭曲、频率掩蔽和时间掩蔽，以促使模型学习更鲁棒的特征并提升其泛化能力。接着，在归一化层中使用均值-方差对特征进行归一化。卷积下采样模块对输入特征进行下采样，从而显著减少后续过程中的计算量。下采样后的特征与位置信息相加，其总和作为编码器层的输入。

编码器：编码器用于从数据预处理后的音频特征中提取高层声学特征。编码器由 L 层组成。在所提出的 Sim-T 模型中，为了分析复用技术在不同权重共享层数量下对模型性能和参数缩减的影响，我们将 L 个编码器层划分为 G 组（$1 \leq G \leq L$）进行研究。具体而言，每组中的层数定义为 L/G。各组对应模块中的权重采用复用方式，而非像 ALBert 网络那样复用网络的所有层。如图 1 中部所示，每个编码器组主要包含三个模块：Pre-MHA、Post-MHA 和前馈神经网络。每个模块都包含残差连接 [32]和层归一化 [33]。每组中的第一个编码器层会构建新的模块权重和注意力分数，后续的 N 个编码器层则复用这些权重和分数，其中$N = L/G - 1$。值得一提的是，当 $G = L$时，Sim-T 中的编码器与原始 Transformer 编码器相同。也就是说，不应用权重复用。 $G = 1$ 表示所有编码器层共享相同的权重。

鉴于编码器输入特征可以表示为 $x \in \mathbb{R}^{T_1 \times d_{model}}$。（$T_1$ 是声学特征的长度； d_{model} 是输入特征的维度。） X_i $\in (X_1, .., X_G)$ 用于表示第 i个编码器组的输出。 X_i^j 用于表示第 i组中的第 j层的输出。 X_i^j 可表示如下：

$$X_i^j = \begin{cases} FFN(O_i) & j = 1, \\ FFN[Post\text{-}MHA(S_i, X_i^{j-1})] & 1 < j \leq (N+1), \end{cases} \tag{1}$$

$$S_i, O_i = Pre\text{-}MHA(X_{i-1}^{N+1}, X_{i-1}^{N+1}, X_{i-1}^{N+1}), \tag{2}$$

其中， $X_0 = x$。 $S_i \in \mathbb{R}^{T_1 \times T_1}$ 和 $O_i \in \mathbb{R}^{T_1 \times d_{model}}$ 表示 Pre-MHA 模块的注意力分数和输出特征，

分别位于第 i组中。

传统的Transformer网络使用 MHA捕获输入序列的全局上下文信息。向量点积和$Softmax$ 产生的注意力分数能够有效地表示上述全局内容。然而，注意力分数的计算复杂度与输入特征的长度呈二次方关系。这意味着它会带来巨大的计算开销和大量参数。此外，相邻层的注意力分数十分相似，表明它们之间存在很高的冗余。因此，在每一层更新注意力分数并不是最优方案。

这里我们提出了两种注意力方法，即 Pre-MHA 和 Post-MHA，如图 1 所示。Pre-MHA 与传统注意力机制类似，将查询向量（q）、键向量（k）和值向量（v）作为输入，然后分别通过三个不同的线性层映射为 Q、 K 和 V，最终得到注意力分数（S）和输出特征（O）。具体而言，Post-MHA 将共享 Pre-MHA 计算得到的注意力分数，并将当前输入与共享的注意力分数进行点积运算，以减少注意力分数计算中的更新次数和整体计算开销。Pre-MHA 和 Post-MHA 的表达式如下：

$$S, O = Pre\text{-}MHA(q, k, v), \tag{3}$$

$$Pre\text{-}MHA(q, k, v) = Concat(Head_1, ..., Head_h)W^o, \tag{4}$$

$$Head_n = ATTENTION(Q, K, V), \tag{5}$$

$$ATTENTION(Q, K, V) = SV, \tag{6}$$

$$S = SOFTMAX(\frac{QK^T}{\sqrt{d_k}}), \tag{7}$$

$$Q = qW_n^q, K = kW_n^k, V = vW_n^v, \tag{8}$$

$$Post\text{-}MHA(S, v) = SvW_n^v, \tag{9}$$

其中，$W^o \in \mathbb{R}^{hd_q \times d_{model}}$、$W_n^q \in \mathbb{R}^{d_{model} \times d_q}$、$W_n^k \in \mathbb{R}^{d_{model} \times d_k}$、$W_n^v \in \mathbb{R}^{d_{model} \times d_v}$、$n \in (1, ..., h)$，以及 $d_q = d_k = d_v = d_{model}/h$。$h$ 表示多头操作的头数。

在 Sim-T 模型的编码器中，Pre-MHA 和 Post-MHA 模块的输出将输入 FFN 模块，该模块主要由两个线性层组成。第一层将模块映射到更高维度，第二层则将模块映射到更低维度。通过这一操作，FFN 的输入和输出能够保持相同的维度，从而高效提取鲁棒特征。

$$FFN(X) = ReLU(XW_1 + b_1)W_2 + b_2, \tag{10}$$

其中 $RELU$ 是激活函数，$W_1 \in \mathbb{R}^{d_{model} \times d_{ff}}$、$W_2 \in \mathbb{R}^{d_{ff} \times d_{model}}$ 是可训练的参数矩阵，d_{ff} 是前馈神经网络的隐藏维度。b_1 和 b_2 是偏置向量。

解码器：解码器对编码器生成的高层声学特征进行解码。在 Transformer 中，编码器的输出 X_G 和相应的标签特征 $y \in \mathbb{R}^{T_2 \times d_{model}}$（$T_2$ 是标签特征的长度）被用作解码器的输入。输入。在解码器中，使用注意力机制计算声学特征和标签特征的一系列上下文相关向量空间表示。然而，在 Transformer 解码器中，随着网络层数加深，标签特征的信息往往会减弱。同时，解码器规模对模型解码速度有显著影响。

在Sim-T 模型中，我们提出了一种新的解码器结构，以利用标签序列之间的交互关系，如图1右侧所示。每个解码器层包含四个模块，即 Pre-MHA模块、MHA模块、Post-MHA模块和FFN模块。每个模块都包含残差连接 [32]和层归一化 [33]。$Y = (Y_1, .., Y_M)$用于表示每个解码器层的输出，Y_i 表示第 i 层的输出（$0 \le i \le M, Y_0 = y$）：

$$Y_i = FFN(Post\text{-}MHA(FFN(MHA(X_G, X_G, O_i)), S_1)), \tag{11}$$

$$S_i, O_i = Pre\text{-}MHA(Y_{i-1}, Y_{i-1}, Y_{i-1}). \tag{12}$$

具体而言，我们引入 Post-MHA 模块，对第一个解码器层中的注意力分数进行复用，即$S_1 \in \mathbb{R}^{T_2 \times T_2}$，使模型能够学习更多标签交互信息。尽管每个解码器层比原始结构多两个模块，但表 IV 中的实验结果表明，这种解码器结构只需两层即可达到 Speech-Transformer 模型六层的相同效果。因此，计算开销和参数数量都能大幅降低。

B. 复用技术

MiniVit [31] 提出的权重复用技术使所有 Transformer 层能够共享相同的权重。与 MiniVit 中使用的方案不同，我们提出的复用技术更加灵活，并且由于引入了分组操作和注意力分数复用，可以扩展到注意力分数领域。该技术首先将网络层划分为若干组，使每组中的层能够复用模块权重和注意力分数。因此，本研究中的复用技术由两部分组成，即模块权重复用和注意力分数复用。

模块权重复用：在神经网络中，模型通常由多个堆叠的层组成。因此，模型规模与模型层的数量成正比。为了扩展模型的应用场景，人们希望获得一个性能相近甚至更好的小型模型。为满足这一需求，我们探索了一种方案：模型规模由设置参数决定，在本研究中，该参数为组的数量，而不是网络层的数量。结果表明，本研究提出的模块复用技术是一种令人满意的解决方案。

特别地，模块权重复用已应用于 Transformer网络，以实现模型压缩。如图1所示，编码器的 L 层被划分为 G 组。每组包含 $N + 1$ 层，其中包括 $N + 1$个前馈网络模块、N 个 Post-MHA 模块和一个 Pre-MHA 模块。

在每个组内，相同模块之间进行权重复用。在 Pre-MHA 模块和 Post-MHA 模块中，向量 V 由线性变换$V = vW^v$生成，其中 W^v也进行复用。在解码器中，每一层的 FFN 模块都进行权重复用。Post-MHA、MHA 和 Pre-MHA 也进行权重复用。只有在输入到下一个编码器组或下一个解码器层时，权重才会更新。

注意力分数复用：使用注意力机制捕获序列的全局上下文，是 Transformer 模型取得卓越性能的重要手段。如 III-A节所述，注意力分数复用可以节省计算量，且对模型性能影响很小。因此，本研究提出并应用了一种称为注意力分数复用的方案。

在编码器中，注意力分数仅在 Pre-MHA 模块的每个组中计算一次。之后，这些分数将在后续 N层中的 Post-MHA模块之间共享。在此过程中， $Q = qW^q$、$K = kW^k$和 S计算可减少 N 倍。因此，第 j层的第 i个编码器组中的 Post-MHA 可表示为：

$$Post\text{-}MHA(S_i, X_i^{j-1}) = S_i V = S_i X_i^{j-1} W^v. \quad (13)$$

注意力分数复用在解码器中的应用不同于其在编码器中的应用。其主要目的是复用标签特征的交互关系 $S_1(Q, K) = SOFTMAX[(yy^T)/\sqrt{d_k}]\ y$。解码器包括 M 个 Post-MHA 模块，这些模块共享相同的注意力分数 S_1。另一个输入 $v_i = MHA(X_G, O_i, X_G)$ 是高层声学特征。因此，第 i个解码器层中的 Post-MHA 模块可表示为：

$$Post\text{-}MHA(S_1, v_i) = S_1 V = S_1 v_i W^v. \quad (14)$$

在注意力分数复用技术的帮助下，每个解码器层都将包含标签特征的交互信息。这确保了模型能够有效地学习有助于改进模型的鲁棒特征。

IV. 实验

A. 实验设置

三个广泛使用的数据集，即AISHELL-1 [34],、HKUST[35] 和WSJ [36], ，被选用来验证所提出的 Sim-T 网络的有效性。需要注意的是，AISHELL-1 和 HKUST 是中文数据集，而 WSJ 是英文数据集。图 1展示了前端对输入音频信号进行的有效预处理，包括分帧、加窗、快速傅里叶变换和离散余弦变换。窗长为 25ms。窗移为 10ms，输出特征为 80 维对数 Mel 滤波器组。

所有实验均采用 ESPNet 工具包 [37] 完成。与 ESPNet 中实现的其他方法类似，Sim-T 模型在 AISHELL-1 数据集上训练 50 个 epoch，在香港科技大学数据集上训练 35 个 epoch，并且

表 I Sim-T 与其他自动语音识别模型在 AISHELL-1 数据集上的比较

模型	开发集 (%)	测试集 (%)
无语言模型的 Transformer [10]	5.5	5.9
带语言模型的 Transformer [10]	5.3	5.6
RNN-T [39]	10.13	11.82
SA-T [39]	8.30	9.30
Chunk-Flow SA-T [39]	8.58	9.80
Sync-Transformer [40]	7.91	8.91
Masked-NAT [41]	6.4	7.1
ESPNet-RNN [42]	6.8	8.0
Insertion-NAT [43]	6.1	6.7
LASO [44]	5.8	6.4
AT [45]	5.5	5.9
带语言模型的 RealFormer [13]	5.7	6.1
不带语言模型的 Sim-T	5.3	5.7
带语言模型的 Sim-T	**4.9**	**5.2**

表 II 香港科技大学数据集上的自动语音识别模型性能。

模型	字符错误率 (%)
Chain-TDNN [46]	23.7
自注意力对齐器 [47]	24.1
Extended-RNA [48]	26.6
联合CTC-注意力模型/ESPNet [49]	27.4
带语言模型的 SAM [47]	24.92
带语言模型的 CTC [50]	34.8
无语言模型的 Transformer [10]	21.7
带语言模型的 Transformer [10]	21.5
无语言模型的 Sim-T	21.6
带语言模型的 Sim-T	**21.2**

在《华尔街日报》数据集上训练 100 个 epoch。采用标签平滑和 dropout 正则化（$p = 0.1$）以防止过拟合。优化器为 Adam [38] ，学习率为 0。002，预热步数为 25000，ϵ 为 10^{-9}、$\beta 1 = 0.9$ 和 $\beta 2 = 0.98$。其他参数为：$d_{model} = 256$、$d_{ff} = 2048$、$h = 4$、$N = 12$ 和 $M = 2$。与注意力维度相关的参数设置为：$d_q = d_k = d_v = d_{model}/h = 64$。每个批次的初始大小设为 64，波束搜索算法中的 $beam\ size$为 10。语言模型（LM）包含 16 层，并采用 Transformer 框架训练 15 个 epoch。

B. 与其他热门模型的比较

如表 I 所示，显然，所提出的 Sim-T 模型在 AISHELL-1 数据集上的表现优于其他热门方法。采用语言模型（LM）时，Sim-T 在开发集和测试集 上的字符错误率（CER）分别为 4.9% 和 5.2%。与基线 Transformer 模型相比，两者的字符错误率均降低了 0.4%。不使用语言模型时，在开发集 和测试集 上取得的字符错误率分别为 5.3% 和 5.7 %，仍比基线低 0.2%。值得注意的是，Sim-T 的参数量仅为 15.62M ； 相比基线 Transformer 模型的 30.36M，减少了 14.74M（48%）参数。

香港科技大学数据集上的实验结果如表 II 所示。文中还列出了一些常见结果以供比较。显然，Sim-T 模型的性能更好，所涉及的参数和计算量也更少。加入和不加入语言模型（LM）时，字符错误率（CER）

表 III 《华尔街日报》数据集上自动语音识别模型的词错误率比较。

模型	dev93 (%)	eval92 (%)
DeCoAR [51]	8.3	4.6
SAN-CTC [52]	8.9	5.9
E2E-SincNet [53]	7.8	4.7
带语言模型的 Transformer [10]	6.6	4.6
带语言模型的 Sim-T	**6.5**	**4.4**

表 IV Sim-T 6×4、Sim-T 6×2 和基线 Transformer 的比较。

模型	开发集(%)	测试集(%)	模型规模(M)
带语言模型的 Transformer	5.3	5.6	30.36
无语言模型的 Transformer	5.5	5.9	-
Sim-T 6×4 与语言模型	4.9	5.3	18.25（41% ↓）
Sim-T 6×4 不使用语言模型	5.5	5.9	-
Sim-T 6×2 与语言模型	**4.9**	**5.2**	**15.62** (48% ↓)
Sim-T 6×2 不使用语言模型	5.3	5.7	-

分别为 21.2% 和 21.6%，比基线 Transformer 网络分别低 0.3% 和 0.1%。基线 Transformer 模型的规模为 29.85M，而 Sim-T 模型为 17.77M，因此参数数量减少了 40%。此外，Sim-T 的性能优于其他常见网络。

此外，我们还在名为 WSJ 的英语数据集上进行了进一步实验，以验证Sim-T网络在不同语言中的性能，结果如表 III所示。与流行网络相比，Sim-T仅使用更少的参数也能取得一定的改进。在 *eval93* 和*eval92* 上的词错误率（WER）分别为 6.5%和4.4%，与基线 Transformer 相比，分别提高了 0.1%和 0.2%。这里，基线 Transformer 的模型规模为 27.14M，但 Sim-T 的模型规模降至 16.28M，参数减少了 40%。

基于上述实验结果，可以得出结论：复用技术对于网络压缩是有益的。此外，在适当的参数设置下，网络性能并未下降，反而得到了令人惊讶的提升。特别是，当将复用技术应用于Transformer 网络时，在模型性能不下降的情况下，参数量至少减少了 40%。

C. 消融研究

为了系统地考察所提出的 Sim-T 模型的有效性，我们对网络的主要组件进行了研究，所有实验均在 AISHELL-1 数据集上完成。为了理解模型，以下一些内容值得探究，例如Sim-T 网络结构的优越性、所提出的 Sim-T 网络中多路复用和分组的影响，以及多路复用的扩展应用。

Sim-T 模型结构的优越性： 选取了三个模型进行实验探究，以证明在使用与 Transformer模型相同或更少数量的模块时，Sim-T 模型在压缩模型情况下的高性能。这些选定的模型是基线

表 V 层归一化权重复用对模型性能和模型规模的影响。

模型	开发集(%)	测试集(%)	模型规模(M)
带语言模型的 Transformer	5.3	5.6	30.36
带语言模型的 Sim-T I	4.9	5.2	15.62 (48% ↓)
带语言模型的 Sim-T II	5.2	5.6	**9.32** (**69%** ↓)

具有十二个编码器层和六个解码器层的 Transformer 模型、具有六个编码器组和四个解码器层的 Sim-T 6×4 模型，以及具有六个编码器组和两个解码器层的 Sim-T 6×2 模型。Sim-T 6×4 和 Sim-T 6×2的每个组中都有两个编码器层。值得一提的是，在Transformer 中，每个解码器层包含一个 FFN模块和两个与 MHA 相关的模块，而在 Sim-T 中，每个解码器层包含两个 FFN 模块和三个与 MHA 相关的模块。为保证公平性，选择 Sim-T 6×4 与 Transformer网络进行比较，因为二者拥有数量相近的对应模块。Sim-T 6×2 中提到的解码器包含更少的模块，用于验证本文提出的新型解码器结构的优越性。

相应的实验结果如表 IV 所示。根据实验结果可以注意到，与基线 Transformer 相比，当采用语言模型时，Sim-T 6×4 在开发集和测试集上的模型性能分别提升了 0.4% 和 0.3% 字符错误率。此外，在不使用语言模型时，该模型的性能与基线相同。需要特别指出的是，Sim-T 6×4 的模型规模为 18.25M，而基线 Transformer 的模型规模为 30.36M，这表明基线中的参数有 41% 得到了压缩。基于上述分析可以发现，所提出的 Sim-T 网络结构优于 Transformer，因为在相应模块数量相近的情况下可以获得更好的性能。

值得注意的是，与 Sim-T 6×4相比，Sim-T 6×2 性能更好且更加轻量。相比之下，Sim-T 6×2 的模型参数量比 Sim-T 6×4减少了 0.7%。使用语言模型时，Sim-T 6×2 在开发集 上的字符错误率比 Sim-T 6×4提升了 0.1%。不使用语言模型时，Sim-T 6×2在开发集 和测试集 上的字符错误率相比 Sim-T 6×4提升了 0.2%。此外，与带语言模型的基线 Transformer 相比，Sim-T 6×2 在开发集 和测试集 上的模型性能均提升了 0.4% 字符错误率。不使用语言模型时，在开发集和测试集上均可实现 0.2% 的字符错误率提升。Sim-T 6×2 的模型规模为 15.62M，与基线相比，其中 48% 的参数可以被压缩。令人惊讶的是，Sim-T 6×2 的性能优于 Sim-T 6×4，这进一步证明了所提出的网络结构对于模块压缩是有效的。

复用技术： 如 III-A节所述，Sim-T 的编码器和解码器中的每个模块都包含残差连接和层归一化。这里我们深入探讨同类型模块中的层归一化参数共享如何影响模型性能和模型规模。例如，在每个编码器组中，

表 VI 在 Transformer 网络的不同部分使用复用技术。

编码器	解码器	开发集（%）	测试集（%）	模型规模(M)
×	×	5.3	5.6	30.36
✓	×	5.3	5.5	17.21 (43% ↓)
×	✓	5.2	5.6	22.47 (26% ↓)
✓	✓	5.2	5.6	9.32 (69% ↓)

表 VII 不同编码器组规模的比较。

模型	开发集(%)	测试集(%)	模型规模(M)
带语言模型的 Transformer	5.3	5.6	30.36
Sim-T 6×2 带语言模型	5.2	5.6	9.32 (69% ↓)
Sim-T 4×2 与语言模型	5.3	5.6	8.97 (71% ↓)
Sim-T 3×2 与语言模型	5.5	5.8	8.52 (71% ↓)
Sim-T 2×2 与语言模型	5.7	6.1	8.28 (72% ↓)
带语言模型的 Sim-T 1×2	6.1	6.6	7.99 (73% ↓)

Pre-MHA 模块与 Post-MHA 模块共享层归一化参数。每个 FFN 模块内部也采用参数共享。在解码器中也进行了类似处理，以进一步减少参数量。未采用和采用复用技术的层归一化网络分别称为 Sim-T I 和 Sim-T II。上述两个版本之间的差异已通过实验验证，结果如表 V 所示。可以看出，尽管在使用语言模型时，Sim-T II 在两个开发集上的字符错误率均提升了 0.1%，但 Sim-T II 的模型规模仅为 9.32M，比基线 Transformer 低 69%。也就是说，在 Sim-T I 的基础上还可以压缩更多参数。实验表明，扩展复用技术有望在保持甚至提升模型性能的同时，使模型更加轻量化。

此外，我们还将所提出的复用技术应用于基线 Transformer 模型，以分别验证其对编码器和解码器的影响。实验结果如表 VI 所示。仅在编码器中应用复用技术时，在模型规模为 17.21M、压缩率为 43% 的情况下，字符错误率相比基线 Transformer 在测试集上降低了 0.1%。另一方面，仅在解码器中应用复用技术时，在模型规模为 22.47M、压缩率为 26% 的情况下，字符错误率相比基线 Transformer 在开发集 上降低了 0.1%。在编码器和解码器中均采用权重复用技术，可将模型规模压缩至 9.32M，并使开发集 上的字符错误率降低 0.1%。因此，可以认为，无论是在编码器还是解码器中应用复用技术，都能在压缩模型的同时不牺牲模型性能。

组数量分析： ALBert [3]采用的跨层权重共享技术，只是简单地共享模型中所有层的权重。Universal Transformer [18] 和 MiniVit [31]也采用了类似的技术。与这两种方法不同，本研究提出的 Sim-T 将所有编码器层划分为多个组，并在每个组中采用所提出的复用技术。对于

表 VIII 语言模型和 SimLM 的模型规模与内存。

模型	PPL	模型规模（M）	内存（G）
语言模型 [9]	49.74	53.24	1.396
SimLM	51.86	**5.94(89% ↓)**	**0.572(59% ↓)**

表 IX 带语言模型的 Transformer 与 SimLM 在 AISHELL-1 数据集上的性能比较。

模型	开发集(%)	测试集(%)
带语言模型的 Transformer	5.3	5.6
带 SimLM 的 Transformer	5.8	6.1

例如，十二个编码器层被划分为四个组，因此每组中的相应模块会复用模块权重和注意力分数。当只有一个编码器组时，这意味着所有层的权重都是共享的。鉴于 Sim-T $i\times2$ 意味着所有编码器层被划分为 i 组，并包含两个解码器层。在此，我们选择 Sim-TII 进行实验验证，结果如表 VII 所示。根据实验结果，编码器组的数量从六个逐渐减少到一个，同时开发集 和 测试集 上的模型性能分别从 5.2% 字符错误率逐渐下降至6.1% 字符错误率，以及从 5.6% 字符错误率逐渐下降至 6.6% 字符错误率。应当强调的是，随着编码器组数量的减少，模型性能会下降。此外，随着编码器组数量的增加，模型规模从 9.32M逐渐减小至 7.99M。从表中可以注意到，尽管能够获得更高程度的模型压缩，但当编码器组数量为一个时，模型性能并不理想。因此，在将复用技-术应用于压缩时，设置适当的组数量是模型性能与规模之间极其重要的权衡。例如，本研究中最适用的编码器组数量是六个。

语言模型的应用： 最后，我们对语言模型进行了扩展实验，以进一步验证所提出的复用技术的压缩效果。用于比较的基线模型是一个基于 Transformer、由 16 个编码器层组成的语言模型（LM）。本研究提出了一种名为 SimLM 的新型语言模型，通过应用复用技术来简化 LM。SimLM 将 16 个编码器层划分为 8 个编码器组，每组包含 2 层。两种语言模型的困惑度（PPL）、模型规模以及训练期间的内存大小如表 VIII 所示。其中，LM 的 PPL 为 49.74，而 SimLM 的 PPL 增至 51.86（增加了 2.12）。然而，LM 的模型规模为 53.24M，而 SimLM 仅为 5.94M。也就是说，参数量压缩了 89%。LM 的内存大小为 1.396G，而 SimLM 降至 0.572G，表明节省了 59% 的内存空间。此外，我们还将 LM 和 SimLM 分别插入 Speech Transformer 网络，进行自动语音识别任务测试，以展示所提出的 SimLM 的作用。实验结果如表 IX 所示。在开发集 和测试集上，采用 SimLM 的 Transformer 模型的字符错误率比采用 LM 的模型低 0.5%。从

表中可以看出，模型性能与参数数量之间存在令人满意的权衡。这是因为模型规模和内存占用显著降低，而模型性能的下降始终可以忽略不计。这进一步证明，所提出的复用技术能够在不严重牺牲模型性能的情况下，大幅压缩模型。

基于上述所有消融研究，本文对所提出的 Sim-T 在多个技术问题上进行了探索和验证。结论如下：首先，在每个解码器层中应用标签特征之间的交互，有助于提升模型性能。通过这种方式，可以使用比基线 Transformer 网络更少的解码器层，同时获得更好的性能。其次，在模块之间复用层归一化（LN）中的参数，可以进一步压缩模型。第三，随着编码器组数量减少，模型规模会变小，但模型性能往往会下降。最后，复用技术可以扩展到语言模型，在不对模型性能造成严重影响的情况下，大幅压缩模型。

V. 结论

本研究提出了一种名为 Sim-T 的新型轻量级自动语音识别（ASR）模型，该模型能够利用复用技术高效简化 Transformer 网络。通过结合模块权重复用、注意力分数复用以及一种新的解码器结构，Sim-T 能够高效减少参数数量，并取得优于原始 Transformer 模型的性能。其有效性已通过在三个广泛使用的数据集（即 AISHELL-1、HKUST 和 WSJ）上开展的大量实验得到验证。此外，权重复用技术还被应用于语言模型，以验证其泛化能力。结果表明，Sim-T 至少能够减少 40% 的模型参数数量，同时几乎不会造成性能下降。

致谢

本研究得到科研基金项目 No.2020A1515111107、No.2021B1515120025 和 No.2021KSYS008 的资助。

参考文献

[1] Vaswani A、Shazeer N、Parmar N 等。“Attention is all you need”，神经信息处理系统进展，2017年，第 5999-6009 页。[2] Devlin J、Chang M W、Lee K 等。“BERT：用于语言理解的双向 Transformer 预训练”，北美计算语言学协会分会，2019年，第 4171-4186 页。[3] Lan Z、Chen M、Goodman S 等。“ALBERT：用于语言表征自监督学习的轻量级 BERT”，国际学习表征会议，2020年。[4] Zafrir O、Boudoukh G、Izsak P 等。“Q8bert：量化的 8 位 bert”，第五届能源高效机器学习与认知计算研讨会——*NeurIPS* 专场（*EMC2-NIPS*），2019年，第 36-39 页。[5] Lin X、Sun S、Huang W 等。“EAPT：用于图像处理的高效注意力金字塔 Transformer”，*IEEE* 多媒体汇刊，第 25 卷，2021年，第 50-61 页。[6] Liu Z、Lin Y、Cao Y 等。“Swin Transformer：使用移位窗口的层次化视觉 Transformer”，*IEEE/CVF* 国际计算机视觉会议论文集，2021年，第 10012-10022 页。

[7] Fan H、Xiong B、Mangalam K 等。《多尺度视觉变换器》，IEEE/CVF国际计算机视觉会议论文集，2021年，第6824-6835页。[8] Dai Z、Yang Z、Yang Y 等。《Transformer-XL：超越固定长度上下文的注意力语言模型》，计算语言学协会第57届年会论文集，2019年，第2978-2988页。[9] Al-Rfou R、Choe D、Constant N 等。《通过更深层的自注意力进行字符级语言建模》，AAAI人工智能会议，2019年，第3159-3166页。[10] Dong L、Xu S、Xu B。《Speech-transformer：用于语音识别的无循环序列到序列模型》，IEEE国际声学、语音与信号处理会议（ICASSP），2018年，第5884-5888页。[11] Wei G、Duan Z、Li S 等。《LFEformer：利用具备可变形能力的滑动窗口进行局部特征增强，以实现自动语音识别》，IEEE信号处理快报，第30卷，2023年，第180-184页。[12] Gulati A、Qin J、Chiu C C 等。《Conformer：用于语音识别的卷积增强型变换器》，Interspeech，2020年，第5036-5040页。[13] He R、Ravula A、Kanagal B 等。《RealFormer：具备残差注意力的Transformer》，ACL-IJCNLP，2021年，第929-943页。[14] Li J、Wang X、Li Y。《用于大规模中文普通话语音识别的speechtransformer》，ICASSP，2019年，第7095-7099页。[15] Zhikui Duan、Guozhi Gao、Jiawei Chen 等。《用于语音识别的双残差Transformer网络》，《音频工程学会期刊》，70(10)，2022年，第871-881页。[16] LeCun Y。《泛化与网络设计策略》，《神经计算》，1989年，第143-155页。[17] Nowlan S J、Hinton G E。《通过软权重共享简化神经网络》，《神经计算》，1992年，第473-493页。[18] Dehghani M、Gouws S、Vinyals O 等。《通用变换器》，ICLR，2019年。[19] Dalvi F、Sajjad H、Durrani N 等。《分析预训练Transformer模型中的冗余》，自然语言处理实证方法会议（EMNLP），2020年，第4908-4926页。[20] Shim K、Choi J、Sung W。《理解自注意力在高效语音识别中的作用》，国际学习表征会议，2021年。[21] Tao F、Busso C。《基于多任务学习的端到端视听语音识别系统》，IEEE多媒体汇刊，第23卷，2020年，第1-11页。[22] Graves A、Fern´andez S、Gomez F 等。《连接主义时序分类：利用循环神经网络标注未分段序列数据》，第23届国际机器学习会议论文集，2006年，第369-376页。[23] Graves A、Mohamed A、Hinton G。《利用深度循环神经网络进行语音识别》，IEEE国际声学、语音与信号处理会议，2013年，第6645-6649页。[24] Malhotra P、Vig L、Shroff G 等。《用于时间序列异常检测的长短期记忆网络》，ESANN，2015年，第89-94页。[25] Chan W、Jaitly N、Le Q V 等。《听、注意与拼写：用于大词汇量会话语音识别的神经网络》，IEEE国际声学、语音与信号处理会议（ICASSP），2016年，第4960-4964页。[26] Kovaleva O、Romanov A、Rogers A 等。《揭示BERT的黑暗秘密》，自然语言处理实证方法会议与第9届自然语言处理国际联合会议（EMNLP-IJCNLP），2019年，第4365-4374页。[27] Jiao X、Yin Y、Shang L 等。《TinyBERT：面向自然语言理解的BERT蒸馏》，载于计算语言学协会论文集：EMNLP 2020，2020年，第4163–4174页。[28] Touvron H、Cord M、Douze M 等。《训练数据高效的图像变换器与基于注意力的蒸馏》，国际机器学习会议，2021年，第10347-10357页。[29] Rao Y、Zhao W、Liu B 等。《Dynamicvit：采用动态令牌稀疏化的高效视觉变换器》，神经信息处理系统进展，2021年，第13937-13949页。[30] Gu J、Wang C、Zhao J。《Levenshtein变换器》，神经信息处理系统进展，2019年。[31] Zhang J、Peng H、Wu K 等。《Minivit：通过权重复用压缩视觉变换器》，计算机视觉与模式识别会议，2022年，第12145-12154页。[32] He K、Zhang X、Ren S 等。《用于图像识别的深度残差学习》，载于IEEE计算机视觉与模式识别会议，2016年，第770-778页。[33] Ba J L、Kiros J R、Hinton G。《层归一化》，arXiv:1607 06450，2016年。

[34]Bu H、Du J、Na X 等。《AISHELL-1：开放源代码普通话语音语料库及语音识别基线》，国际语音数据库与语音输入/输出系统及评估协调委员会东方分会会议，2017年，第1-5页。[35] Liu Y、Fung P、Yang Y 等。《HKUST/MTS：超大规模普通话电话语音语料库》，中文口语处理国际研讨会，2006年，第724-735页。[36] Paul D B、Baker J。《基于《华尔街日报》的CSR语料库设计》，语音与自然语言研讨会论文集，1992年，第357–362页。[37] Watanabe S、Hori T、Karita S 等。《ESPnet：端到端语音处理工具包》，*Interspeech*，2018年，第2207-2211页。[38] Kingma D、Ba J L。《Adam：一种随机优化方法》，国际学习表征会议，2015年。[39] Tian Z、Yi J、Tao J 等。《用于端到端语音识别的自注意力转导器》，*Interspeech*，2019年，第4395-4399页。[40] Tian Z、Yi J、Bai Y 等。《用于端到端语音识别的同步变换器》，*IEEE*国际声学、语音与信号处理会议，2020年，第7884-7888页。[41] Chen N、Watanabe S、Villalba J 等。《用于语音识别的非自回归Transformer》，*IEEE*信号处理快报，2021年，第28卷，第121-125页。[42] Karita S、Chen N、Hayashi T 等。《语音应用中Transformer与RNN的比较研究》，*IEEE*自动语音识别与理解研讨会，2019年，第449-456页。[43] Fujita Y、Watanabe S、Omachi M 等。《用于端到端自动语音识别的基于插入的建模》，*Interspeech*，2020年，第3660-3664页。[44] Bai Y、Yi J、Tao J 等。《专注聆听，一次拼写：通过非自回归架构实现整句生成，以进行低延迟语音识别》，*Interspeech*，2020年，第3381-3385页。[45] Fan R、Chu W、Chang P 等。《CASS-NAT：基于CTC对齐的单步非自回归Transformer语音识别模型》，*IEEE*国际声学、语音与信号处理会议，2021年，第5889-5893页。[46] Povey D、Peddinti V、Galvez D 等。《基于无格MMI的纯序列训练神经网络，用于自动语音识别》，*Interspeech*，2016年，第2751-2755页。[47] Dong L、Wang F、Xu B。《自注意力对齐器：结合自注意力网络与分块跳跃的延迟控制端到端自动语音识别模型》，*IEEE*国际声学、语音与信号处理会议，2019年，第5656-5660页。[48] Dong L、Zhou S、Chen W 等。《扩展循环神经对齐器，实现普通话流式端到端语音识别》，*Interspeech*，2018年，第816-820页。[49] Kim S、Hori T、Watanabe S 等。《基于多任务学习的CTC-注意力联合端到端语音识别》，*IEEE*国际声学、语音与信号处理会议，2017年，第4835-4839页。[50] Miao Y、Gowayyed M、Na X 等。《CTC声学模型的实证探索》，*IEEE*国际声学、语音与信号处理会议，2016年，第2623-2627页。[51] Ling S、Liu Y、Salazar J 等。《用于半监督语音识别的深度上下文化声学表征》，*IEEE*国际声学、语音与信号处理会议，2020年，第6429-6433页。[52] Salazar J、Kirchhoff K、Huang Z。《用于语音识别中连接主义时序分类的自注意力网络》，*IEEE*国际声学、语音与信号处理会议，2019年，第7115-7119页。[53] Parcollet T、Morchid M、Linares G。《E2E-SINCNET：迈向完全端到端语音识别》，*IEEE*国际声学、语音与信号处理会议，2020年，第7714-7718页。