arXiv:2303.03267v1 [cs.CL] 2 Mar 2023

评估 SURE 语音理解基准上的参数高效迁移学习方法

Yingting Li[1,2] *Ambuj Mehrish*[2] *Rishabh Bhardwaj*[2] *Navonil Majumder*[2] *Bo Cheng*[1] *Shuai Zhao*[1] *Amir Zadeh*[4] *Rada Mihalcea*[3] *Soujanya Poria*[2]

[1] 北京邮电大学，中国 [3] 密歇根大学，美国 [2] 新加坡科技设计大学，新加坡 [4] Amazon Science，美国

摘要

微调广泛用作从预训练模型进行迁移学习的默认算法。然而，在迁移学习过程中，如果需要针对各个下游任务更新大型预训练模型的所有参数，就可能造成参数效率低下。随着参数数量增加，微调容易导致过拟合和灾难性遗忘。此外，当模型用于许多任务时，全量微调的成本可能高得难以承受。为缓解这一问题，人们提出了参数高效迁移学习算法，例如适配器和前缀微调，通过引入少量可训练参数，将其插入 BERT 和 HuBERT 等大型预训练语言模型。本文介绍了用于各类语音处理任务的参数高效学习基准——**语音理解**评测（SURE）。此外，我们还提出了一种基于一维卷积的新型适配器 ConvAdapter。结果表明，ConvAdapter 的表现优于标准适配器；在 SURE 的部分任务中，仅使用 0.94% 的可训练参数，其性能与前缀微调和 LoRA 相当。我们还进一步探究了参数高效迁移学习在文本转语音（TTS）等语音合成任务中的有效性。

索引词— 迁移学习、适配器、前缀调优、LoRA、ConvAdapter、语音处理。

1. 引言

用于自然语言处理（NLP）的大型预训练语言模型（PLM）已成为主流范式，并在各种语言理解任务中取得了最先进的结果。尽管针对特定任务对 PLM 进行微调已取得显著成功，但由于模型参数数量庞大，这不仅需要大量计算资源进行训练，还需要专用的图形内存。此外，在特定任务数据有限的情况下，微调容易导致记忆（即过拟合）和灾难性遗忘 [1]。在 NLP 领域，近期研究提出了多种参数高效的迁移学习方法，例如适配器调优 [2], 前缀微调 [3], 低秩适配（LoRA）[4], 和提示微调 [5], ，这些方法仅训练一小部分模型参数，同时避免与完整模型微调相关的问题，即可取得最先进的结果。在下游任务中冻结 PLM 的大量参数后，参数高效方法旨在仅训练少量额外的特定任务参数，从而减轻灾难性遗忘 [6], ，并免除额外内存和计算资源的需求。与 NLP 和计算机视觉（CV）相比，适配器在语音领域的应用迄今仅限于少数涉及自动语音识别（ASR） [7], 、跨语言/多语种自动语音识别 [8] 和语音翻译 [9]的研究。鉴于参数高效方法在各种下游 NLP 任务中取得了无可争辩的成功，评估其解决下游语音处理任务的有效性至关重要。由于解决 NLP 任务和语音处理任务的方法具有许多共同特征，我们假设 NLP 中的参数高效算法能够在语音处理中带来类似的收益。

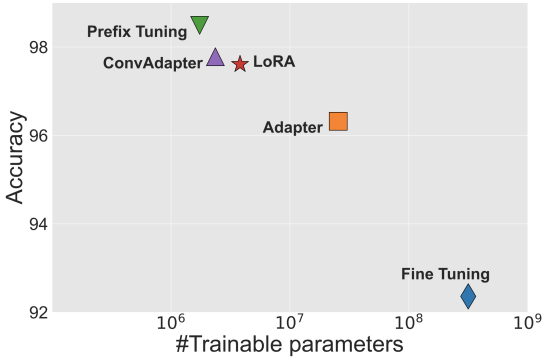


图 1。在 VCTK 数据集上进行说话人识别任务时，不同参数高效方法与全量微调的准确率（%）和可训练参数量对比。

语音处理领域的大型预训练语言模型（如 Wav2Vec 2.0 [10], 、HuBERT [11], 和 WaveLM [12]）已发展到能够对语音信号进行有效编码，生成表征，这一点体现在

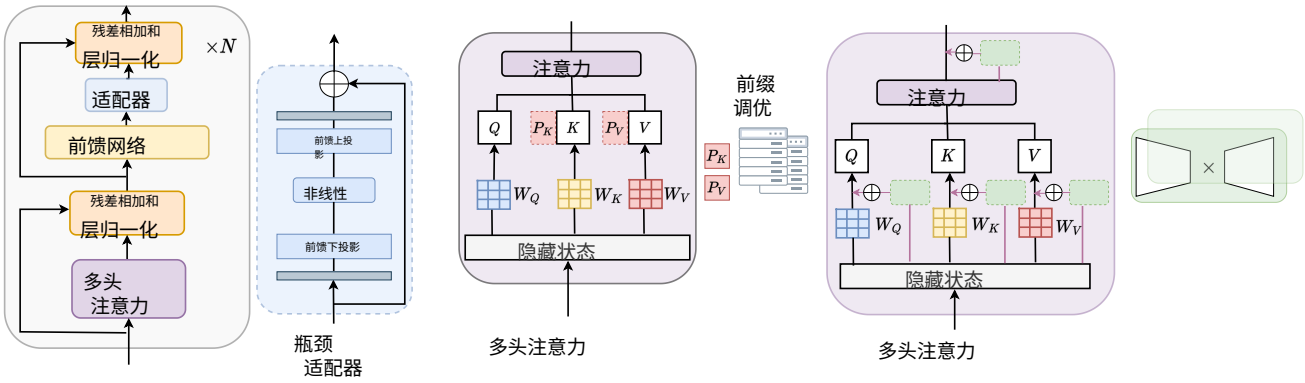


图 2。Transformer 架构以及适配器、前缀微调和 LoRA。

下游语音处理任务的最先进性能上 [13, 14]。尽管取得了良好进展，针对不同下游任务对这些模型进行完全微调仍面临多重挑战：i）需要大规模的特定任务数据集；ii）需要微调模型的所有参数，并为每个下游任务分配专用内存。例如，在情感分析中，由于数据收集和标注耗时且成本高昂，训练数据量通常较少，标签也存在噪声。此外，Wav2Vec 2.0 的最大版本包含 3.17 亿个参数。鉴于自然语言处理和语音预训练语言模型在微调方面面临的挑战相似，我们认为，自然语言处理领域的参数高效方法能够惠及语音处理下游应用。因此，我们的贡献如下：

[贡献 1] 我们针对不同的语音处理任务（如语音情感识别（SER）、说话人识别（SR）、自动语音识别（ASR）和关键词检出（KS））， 提供了一套有效的最先进参数高效方法配置。为此，我们引入了一个新的基准 SURE，用于检验不同参数高效方法（即适配器调优、前缀微调和 LoRA）的能力。

在语音领域，可以利用卷积神经网络（CNNs）进一步挖掘语音信号相邻时频分量之间的强相关性。这促成了我们的第二项贡献。

[贡献 2] 我们提出了一种基于卷积神经网络的新型轻量级适配器 ConvAdapter。尽管本文使用的是 Wav2Vec 2.0，ConvAdapter 也能很好地扩展到任何基于 Transformer 的语音模型，包括 HuBERT [11]。我们希望本文呈现的结果和分析能进一步推动更具泛化能力、可复用的参数高效方法的发展，从而促进语音研究。

2. 相关工作

为探索大型预训练语言模型在各种语音处理任务中的能力，Yang 等人 [15] 提出了 SUPERB 基准。他们冻结预训练的自监督学习（SSL）模型，并在模型之上添加一个特定于任务的预测层。这种方法是在下游输入中加入与任务相关的描述，以帮助预训练模型理解下游任务。然而，这种方法未能探索上游模型的能力。

适配器最初作为用于迁移学习的参数高效算法被引入自然语言处理领域 [16]。在各种自然语言处理任务中，适配器已展现出显著成效：只需使用少得多的可训练参数，便能达到与全量微调相当的性能。适配器的优势也已在计算机视觉任务中得到证实 [2]。适配器在语音处理中的应用主要局限于自动语音识别 [7, 17]。近来，适配器也开始用于语音翻译，但探索范围有限。 [18] 研究了一个非常具体的场景（零样本语音翻译）， 而 [9]仅在 Transformer 编码器之后使用了单个适配器。循环神经网络转导器（RNN-T） [8] 架构为流式端到端多语言系统引入了一个适配器模块。 [19], 阐述了适配器在持续学习中的有效性，作者报告称，在新语言任务上对模型进行预训练所需的时间显著减少，同时不会遗忘先前学到的表征。此外， [20] 的作者探索了将语音提示用于生成式口语语言模型 [21]。

3. 参数高效迁移学习

接下来，我们将详细介绍参数高效迁移学习中考虑的不同方法。图2和图3展示了这些不同方法。

适配器调优。 A（瓶颈型或标准型）适配器 [16,22]是一种神经模块，其参数数量远少于语言模型。它将输入 $h \in \mathbf{R}^d$ 下投影到一个 m 维空间（$m < d$），并应用非

线性变换 $g(\cdot)$，最后将结果上投影回 d 维。添加残差连接以获得最终输出，如下所示

$$\boldsymbol{h} \leftarrow \boldsymbol{h} + g(\boldsymbol{h}\boldsymbol{W}_{\text{down}})\boldsymbol{W}_{\text{up}} \tag{1}$$

其中，$\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d\times m}$ 和 $\boldsymbol{W}_{\text{up}} \in \mathbb{R}^{m\times d}$ 分别是下投影矩阵和上投影矩阵。经验上发现，带有瓶颈结构的两层前馈神经网络效果良好。尽管适配器有许多变体，我们在实验中采用了 [22]中的设置。如图2所示，适配器被插入到每个 Transformer 模块的前馈层之后。

前缀微调。 它通过在每一层预训练多头注意力的键和值前添加 k 可学习向量，改变 Transformer 的注意力模块。如图2所示，两组可学习前缀向量 $\boldsymbol{P}_K$ 和$\boldsymbol{P}_V$ 分别与原始键 $\boldsymbol{K}$ 和值 $\boldsymbol{V}$拼接，而查询 $\boldsymbol{Q}$ 保持不变。随后，新的键和值用于多头注意力。多头注意力的头$_i$ 表示为：

$$\text{head}_i = \text{Attn}(\boldsymbol{Q}\boldsymbol{W}_Q^{(i)}, [\boldsymbol{P}_K^{(i)}, \boldsymbol{K}\boldsymbol{W}_Q^{(i)}], [\boldsymbol{P}_V^{(i)}, \boldsymbol{V}\boldsymbol{W}_Q^{(i)}]) \tag{2}$$

其中，Attn(·)是缩放点积注意力，定义如下：

$$\text{Attn}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d_k}})\boldsymbol{V} \tag{3}$$

在训练过程中，只有前缀向量 $\boldsymbol{P}_K$ 和 $\boldsymbol{P}_V$ 会被更新。由于前缀微调会修改每一层中的注意力头，因此它很可能能够更好地控制从一层传递到另一层的声学信息，从而有效激发预训练模型中的知识。

LoRA。由 [4], 提出，它通过将可训练的低秩矩阵注入 Transformer 层，来近似权重更新。对于预训练权重矩阵 $W \in \mathbb{R}^{d\times k}$，其权重更新表示为低秩分解 $\boldsymbol{W}+\Delta\boldsymbol{W} = \boldsymbol{W}+\boldsymbol{W}_{\text{down}}\boldsymbol{W}_{\text{up}}$，其中 $\boldsymbol{W}_{\text{down}} \in \mathbb{R}^{d\times r}$、$\boldsymbol{W}_{\text{up}} \in \mathbb{R}^{r\times k}$ 是可调参数，r 是分解矩阵的秩，且满足 $r < d$。对于输入 $\boldsymbol{x}$ 经过多头注意力层中的线性投影时，LoRA 将投影输出 $\boldsymbol{h}$ 修改为

$$\boldsymbol{h} \leftarrow \boldsymbol{h} + s \cdot \boldsymbol{x}\boldsymbol{W}_{\text{down}}\boldsymbol{W}_{\text{up}} \tag{4}$$

在本文中，我们在多头注意力层的四个位置加入 LoRA，如图2所示。由于其非常轻量，可以针对不同任务将许多小型模块添加到预训练模型中，并通过替换模块来有效切换任务。此外，与完全微调的模型相比，LoRA不会引入推理延迟，并且大致能够收敛到训练原始模型 [4]的效果。

卷积适配器（本文提出）。卷积神经网络在语音处理系统中表现出色且十分有价值。过去的许多研究已经验证了卷积神经网络对于语音的有效性。它能够在局部感受野内融合各通道信息，同时学习任务特定信息。因此，它可以有效提取预训练模型参数中编码的知识。受这一特性的启发，我们提出了一种基于卷积神经网络层的轻量级适配器，用于学习任务特定表示。我们使用三个轻量级的一维卷积层来实现该适配器，卷积核大小分别为 k 3 和 5（其中 $k = 5$ 为逐通道卷积 [23]），并结合层归一化和挤压与激励模块[24]。图 3 展示了所提出适配器的架构。本文后续将该适配器称为 *ConvAdapter* 。我们使用逐通道卷积，因为与标准卷积相比，它所需的参数更少，计算效率也更高。最后，通过添加残差连接得到输出。如图 2 所示，ConvAdapter 被插入到与瓶颈适配器相同的位置。

4. 实验与结果

4.1. SURE 基准

为评估这些方法，我们针对四种不同的语音处理任务及其多样化数据集建立了一个基准，称为 **S**peech**U**nde**R**standing **E**valuation（SURE）。表 1 简要介绍了各项任务以及 SURE 中的数据集。评估指标方面，SER、SR、IC 和 KS 使用准确率；ASR 和 PR 则分别使用词错误率（WER%）和音素错误率（PER%）。语音语言理解（SLU）任务（如 SF）同时涉及槽位类型和槽位值 [14]。因此，槽位类型 F1分数和槽位值字符错误率 [25]被用作评估指标。TTS 的结果同时采用客观评估和主观评估呈现（评估所用指标的详细信息见第 4.4 节）。

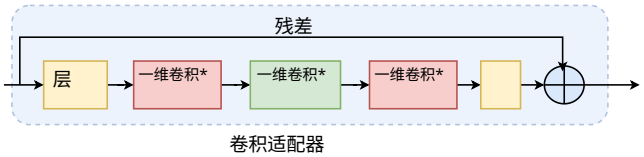


图 3. 基于一维卷积层的轻量级适配器架构。 k 是一维卷积的卷积核大小。
$*$ 表示逐通道卷积。

4.2. 训练

我们采用标准的 Wav2Vec 2.0 架构（XLSR-53-英语），其中包含 24 个 Transformer 模块，并使用 Hugging Face 公开提供的预训练检查点[1]，该检查点经过微调

[1]https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-english

任务	Type	数据集	N
语音情感识别（SER）	识别	ESD [17], MELD[26]	5, 7
说话人识别（SR）	识别	ESD [17], VCTK[27]	10, 110
自动语音识别（ASR）	生成	ESD [17], FLEURS [28], LS [29]	33, 37
音素识别（发音识别）	识别	LS [29]	11
意图分类（意图分类）	口语语言理解	Fluent Speech Commands[25]	5
槽位填充 (SF)	口语语言理解	SNIPS[30]	11
文本转语音（TTS）	生成	LTS [29], L2ARCTIC [31]	247,24
关键词检出（KS）	分类	Speech Commands[32]	11

表 1。SURE 基准中的任务。 N 是每个类别中的类别数（自动语音识别的词汇量）。CLS：分类，SG：序列生成。对于文本转语音， N 表示每个数据集中的说话人数。

方法	#参数	语音情感识别（准确率 % / 加权 F1 分数）↑		语音识别（准确率 %）↑		自动语音识别（词错误率）↓			关键词检出（准确率 %）↑
		ESD	MELD	ESD	VCTK	ESD	FLEURS	LS	语音指令
微调	315,703,947	**96.53**	42.93	99.00	92.36	0.2295	**0.135**	**0.0903**	99.08
适配器	25,467,915 (8.08%)	94.07	41.58	98.87	96.32	0.2290	0.214	0.2425	99.19
前缀微调	1,739,787 (**0.55%**)	90.00	44.21	**99.73**	**98.49**	**0.2255**	0.166	0.1022	98.86
LoRA	3,804,171 (1.20%)	90.00	**47.05**	99.00	97.61	0.2428	0.149	0.1014	98.28
ConvAdapter	2,952,539 (0.94%)	91.87	46.30	99.60	97.61	0.2456	0.2062	0.2958	**98.99**

表 2。SURE 基准：预训练Wav2Vec 2.0 上的全量微调及其他参数高效训练方法。#参数表示 KS 任务的可训练参数数量。括号中的百分比表示可训练参数相对于全量微调的比例。最佳性能以粗体标出，次佳性能以下划线标出。由于数据不平衡，我们在 MELD 上使用加权F1值（简称 w-f1）作为指标。

在 Common Voice 6.1 的英语训练集和验证集上对 XLSR-53 进行微调。XLSR-53 使用 Common Voice、BABLE 和多语言LibriSpeech进行训练。如图 2 所示，我们在每个 Transformer 层中添加适配器。最后，在预训练模型的最后一层后附加一个全连接层，以生成特定任务的输出[2]。对于每项任务，我们将数据集划分为训练集、验证集和测试集。由于数据有限，在适应新任务时，训练过多轮可能会导致过拟合。为避免这种情况，我们采用早停策略。如果验证集上的性能连续五轮没有提升，我们就停止训练，并保存最佳检查点，以便在测试集上进行评估。

4.3. 结果与讨论

表 2 和表 3 表明，在各种语音处理任务中，所研究的参数高效方法表现与全量微调相当或更优。它们在 KS 和自动语音识别上的结果与微调相当。在 ESD数据集上，这些方法的表现大幅优于微调。即使在 MELD 数据集上，其性能也优于微调。不过，由于 MELD 数据集是用于情感识别的多模态数据集，且我们在本研究中只使用单一模态。在说话人识别任务中，我们观察到，采用这些调优范式学习到的表征能够有效识别说话人，即使面对带有情感且带口音的语音也是如此。此外，要取得这些结果所需的可训练参数数量远低于微调——适配器和 ConvAdapter 所需的可训练参数分别仅为全量微调的 8.08% 和 0.94%。图 1 对比展示了说话人识别任务中可训练参数数量与性能的关系。

所提出的 ConvAdapter[3] 的性能与其他适配器和微调方法相近，或略逊一筹。不过，这种轻微的性能损失换来了任务特定可训练参数数量的大幅减少。与微调相比，ConvAdapter 的可训练参数仅为 0.94%。ConvAdapter 的性能与标准适配器和 LoRA 相近，但所需参数显著更少。然而，前缀微调在性能和可训练参数数量方面均优于 ConvAdapter（与微调相比，仅为 0.55%）。由于前缀微调通常会修改每个 Transformer 模块中各注意力头的输出，因此它比其他适配器更具表达能力。

除了表 2 和表 3 中列出的任务外，我们

[2]代码见 https://github.com/declare-lab/ speech-adapters

[3]ConvAdapter 的自动语音识别和槽位填充结果仍需进一步调优，以找到最优超参数。

还将实验扩展到语音合成。下一节将说明在 SURE 基准上讨论的参数高效训练方法在文本转语音（TTS）任务中的有效性。

4.4. 文本转语音（TTS）

文本转语音（TTS），也称为语音合成，旨在将文本合成为清晰、自然的语音。现代 TTS 系统由复杂的神经网络构成，可将自然语言建模过程转换为可听见的语音输出 [33, 34, 35]。基于神经网络的 TTS 系统不断发展，显著提升了合成语音的质量。基于神经网络的 TTS 系统包含三个基本构建模块，分别用于文本分析、声学建模和波形生成。

我们采用基于 Transformer 的 TTS [36] 作为基础模型。Transformer TTS 使用 Transformer [37] 和 Tacotron2[33]作为构建模块。Transformer TTS 的详细架构见第 7 节。在本研究中，我们在 VCTK [27] 数据集上对多说话人 Transformer TTS 进行预训练。为使用参数高效方法适配基础模型，我们将实验重点放在公开可用的 LibriTTS [29]和 L2ARCTIC [17]数据集上。我们首先对基础模型训练 900k 步。适配期间，我们完全冻结基础模型，仅更新额外添加模块（Adapter、Prefix、LoRA 和 ConvAdapter）的参数。我们对模型进行额外 40k 步微调，并为每位说话人验证集中的文本样本合成语音，再从以下方面评估合成语音。

4.4.1. 客观评估

客观评估通过比较生成样本与真实参考样本的声学特征失真来进行。这些评估得出的分数不一定与人类判断相关。评估指标包括梅尔倒谱失真（MCD）分数和词错误率（WER）。MCD用于衡量两段梅尔倒谱序列之间的差异，而WER可作为比较不同语音合成模型的指标，用于衡量它们所生成语音的可懂度。

- 梅尔倒谱失真（MCD）：MCD [38] 可使用合成（目标）语句和参考语句的梅尔频谱图，按照公式5计算。

$$MCD[dB] = \frac{10}{\ln 10}\sqrt{2\sum_{i=1}^{24}(m_{k,i}^{t} - m_{k,i}^{r})^2} \quad (5)$$

其中 $m_{k,i}^{t}$ 和$m_{k,i}^{r}$分别是目标语句和参考语句的梅尔倒谱系数（MCEPs），
地用于 k^{th} 帧。

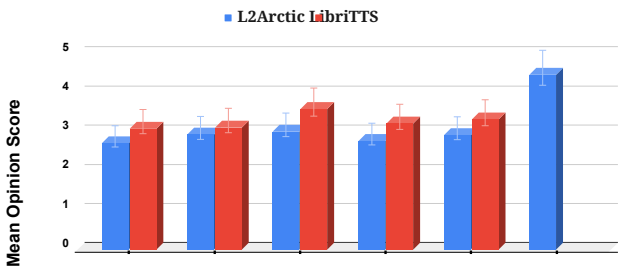


图4。不同参数高效方法的自然度。我们展示了验证集上的结果，所有模型均训练了40k步。

- 词错误率（WER）：在我们的实验中，我们使用企业级语音转文本（STT）预训练 Silero模型 [39]来计算词错误率。我们针对真实语句计算 WER，以便对不同模型进行相对比较。

从梅尔倒谱失真和词错误率来看，适配器在 LTS 数据集上的表现优于其他方法，而 ConvAdapter 在 L2ARCTIC 数据集上的表现优于 LoRA 和前缀微调。性能差异可归因于两个数据集的口音不同：L2ARCTIC 是包含英语母语口音以外多种口音的带口音数据集，而 LTS 主要包含英语母语口音。得益于其架构，ConvAdapter 比前缀微调和 LoRA 更能适应口音变化。

4.5. 主观评估

为了对模型生成的样本进行主观评估，我们进行了平均意见分（MOS）测试，评估自然度和说话人相似度。

- 自然度：为评估自然度，我们为所有说话人合成验证集中未见文本的语音样本，并请听众根据测试的测量目标，按照五点评分自然度量表（1：差，2：较差，3：一般，4：良好，5：优秀）为每段音频话语评分。
- 说话人相似度：为评估合成语音与目标说话人/参考说话人的相似度，我们进行了说话人相似度评估。我们向听众同时呈现目标说话人的真实语音以及由不同方法生成的语音样本，并要求听众参照真实语音 utterance，在 5 点说话人相似度量表上（类似于自然度）对每条语音 utterance 进行评分。

研究人员从事自然语言处理的不同领域，共收集到 9 份主观评估反馈。

方法	IC		PR		SF		
	#参数	FS 准确率 ↑	#参数	LS 音素错误率 ↓	#参数	SNIPS F1 分数 % ↑	字符错误率 ↓
微调	315707288	99.60	311304394	0.0577	311375119	93.89	0.1411
适配器	25471256 (8.06%)	99.39	25278538 (8.01%)	0.1571	25349263 (8.14%)	92.60	0.1666
前缀微调	1743128 (0.55%)	93.43	1550410 (0.49%)	0.1598	1621135 (0.50%)	62.32	0.6041
LoRA	3807512 (1.20%)	99.68	3614794 (1.16%)	0.1053	3685519 (1.18%)	90.61	0.2016
ConvAdapter	3672344 (1.16%)	95.60	3479626 (1.11%)	0.1532	3550351 (1.14%)	59.27	0.6405

表 3. SURE 基准：预训练Wav2Vec 2.0 上全量微调及其他参数高效训练方法在 **FS**：Fluent Speech [25] 和 **LS**：LibriSpeech [29] 数据集上的意图分类和发音识别任务表现。

方法	参数 (%)	LTS		L2ARCTIC	
		梅尔倒谱失真 ↓	词错误率 ↓	梅尔倒谱失真 ↓	词错误率 ↓
微调	35802977	6.2038	0.2655	6.71469	0.2141
适配器	659200	6.1634	0.3143	6.544	0.2504
前缀	153600	6.2523	0.3334	7.4264	0.3244
LoRA	81920	6.8319	0.3786	7.0698	0.3291
ConvAdapter	108800	6.9202	0.3365	6.9712	0.3227

表 4. 使用不同方法生成的样本的梅尔倒谱失真和词错误率。

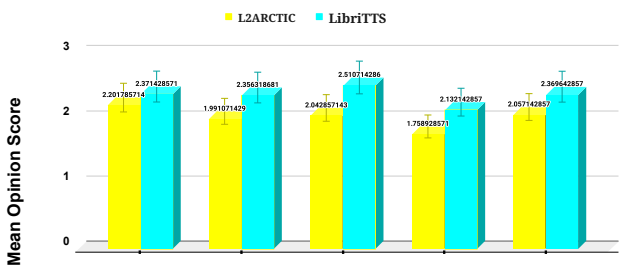


图 5。说话人相似度，不同的参数高效方法。

每位参与者随机聆听从生成样本中选出的 180 个样本。（测试模板和说明请参见附录。）图 4 和图 5 展示了自然度和说话人相似度的得分及其 95% 置信区间。有趣的是，参数高效方法在自然度和说话人相似度方面都取得了与全量微调相近的平均意见分。我们还观察到，微调方法在说话人相似度方面略胜一筹。我们鼓励读者聆听这些语音样本，以了解各方法之间的差异[4]。

[4]https://declare-lab.github.io/speechadapters/

4.6. 讨论

使用少量可训练参数即可获得显著的性能提升，这可能是因为在学习任务专用适配器时，梯度可以沿着预训练模型一路反向传播，从而提炼出其参数中编码的丰富知识。我们提出的 ConvAdapter 仅使用一小部分参数即可实现更好的性能，因此提供了一种参数高效的迁移学习方法。由于 ConvAdapter 仅需要 0.94% 的可训练参数，因此将其与其他适配器（例如前缀微调或 LoRA）结合以进一步提升结果，将会是一个有趣的研究方向。尤其是对于 MELD [26], 这类新的、具有挑战性的数据集，其性能仍不理想。表 2 中报告的结果表明，在预训练自监督学习模型中使用适配器是一个很有前景的未来研究方向。未来，我们可以将基准扩展到不同的基于 Transformer的语音模型，并应用于语音翻译、语音转换和文本转语音等任务。此外，在多语言环境下研究这些模型所学习的表征仍有很大的空间。由于 Wav2Vec 2.0-XLSR-53 是在 53 种不同语言上训练的，因此评估这些模型所学习的跨语言表征将会非常有意义。

先前关于参数高效微调的工作讨论了适配器大小可能带来的性能代价 [16]。较小的适配器引入的参数更少，但可能导致性能下降。因此，为了研究适配器大小对性能的影响，我们还

顶部 k 层	k	24	16	8	4
微调	#参数	315703690	214933898	114164106	63779210
	性能	0.9900	0.9967	0.9980	0.9953
压缩率	n (2^n)	1（2）	2（4）	3（8）	4（16）
瓶颈	#参数	25467658	12878602	6584074	3436810
	性能	0.9887	0.9667	0.9093	0.8313
ConvAdapter	#参数	3668746	3638026	3622666	3614986
	性能	0.9960	0.9947	0.9967	0.9973

表 5. 微调、瓶颈和 ConvAdapter 的验证集准确率及可训练参数数量。瓶颈和 ConvAdapter 使用大小为 2^n 的参数进行调优，取值为 $n = 1$，2、3、4。微调时，对前 k 层进行微调，取值为 $k = 24$，16、8、4。

进行了一项参数量与性能之间的权衡实验。按照 [16]，中所述的设置，我们考虑了不同的适配器大小，并将其与仅微调 Wav2Vec 2.0 顶部 k 层的方案进行了比较。表 5 展示了说话人识别任务中参数量与性能之间的权衡。我们可以从表 5 中观察到，改变压缩率并不会影响 ConvAdapter 的性能，而瓶颈的性能则会下降。微调的结果尤其有趣，因为其性能没有下降。可能的原因是，说话人信息编码在 wav2vec2 模型的初始层中。这些结果凸显了 ConvAdapter 在语音分析中的参数稳健性。尽管 ConvAdapter 的性能与其他方法相近，但它为在各种语音处理任务中有效微调大型预训练模型提供了一种稳健的方法。这项分析进一步揭示了探索大型预训练模型不同层中所编码信息的可能性。通过仅针对特定任务微调特定层，可以进一步利用这些信息，使训练更加参数高效。

5. 结论与未来工作

在这项工作中，我们介绍了 SURE 基准，并探讨了不同的参数高效的基于 Transformer 的适配器在该基准上的有效性。我们开展了大量实验，研究这些方法在广泛的语音处理任务中的能力。实验结果表明，参数量较少的适配器能够达到与全量微调相当甚至更好的性能。最后，我们提出了 ConvAdapter，它能够以有限数量的可训练参数实现最先进的性能。

尽管参数高效方法在文本转语音（TTS）任务上能够取得与微调相近的结果，但仍有改进空间。在实验过程中，我们注意到，尝试生成较长句子时，性能并不理想。此外，语音样本的质量也有进一步提升的空间。未来，我们可以探索如何在数据有限且资源受限的条件下训练高质量的文本转语音模型。我们希望这项工作能推动未来对语音相关任务的参数高效算法的研究。

6. 参考文献

[1] Chuanqi Tan、Fuchun Sun、Tao Kong、Wenchang Zhang、Chao Yang 和 Chunfang Liu，“深度迁移学习综述”，载于国际人工神经网络会议。Springer，2018，第 270–279 页。[2] Sylvestre-Alvise Rebuffi 等，“利用残差适配器学习多个视觉领域”，神经信息处理系统进展，第 30 卷，2017。[3] Xiang Lisa Li 和 Percy Liang，“前缀微调：优化用于生成的连续提示”，*arXiv* 预印本 *arXiv:2101.00190*，2021。[4] Edward J Hu 等，“LoRA：大型语言模型的低秩适配”，载于国际学习表征会议，2021。[5] Brian Lester、Rami Al-Rfou 和 Noah Constant，“参数高效提示微调的规模效应”，载于自然语言处理经验方法会议论文集，2021，第 3045–3059 页。[6] Steven Vander Eeckt 等，“利用适配器克服端到端自动语音识别中的灾难性遗忘”，*arXiv* 电子印本，第 arXiv–2203 页，2022。[7] Bethan Thomas、Samuel Kessler 和 Salah Karout，“自监督语音的高效适配器迁移

用于自动语音识别的模型”，载于*IEEE* 国际声学、语音与信号处理会议（*ICASSP*）。IEEE，2022，第 7102–7106 页。

[8] Anjuli Kannan 等人，《采用流式端到端模型的大规模多语种自动语音识别》，*Interspeech 2019* 会议录，第 2130–2134 页。

[9] Xian Li 等人，《通过对预训练模型进行高效微调实现多语种语音翻译》，载于 *ACL*，2021 年。

[10] Alexei Baevski、Yuhao Zhou、Abdelrahman Mohamed 和 Michael Auli，《wav2vec 2.0：一种用于自监督语音表征学习的框架》，神经信息处理系统进展，第 33 卷，2020 年。[11] Wei-Ning Hsu 等人，《HuBERT：通过隐藏单元掩码预测进行自监督语音表征学习》，*IEEE/ACM*音频、语音与语言处理汇刊，第 29 卷，第 3451–3460 页，2021 年。[12]Sanyuan Chen、Chengyi Wang、Zhengyang Chen、Yu Wu、Shujie Liu、Zhuo Chen、Jinyu Li、Naoyuki Kanda、Takuya Yoshioka、Xiong Xiao 等人，《WavLM：面向全栈语音处理的大规模自监督预训练》，*IEEE*信号处理精选主题期刊，第 16 卷，第 6 期，第 1505–1518 页，2022 年。[13] Alexei Baevski 和 Abdelrahman Mohamed，《自监督预训练对自动语音识别有效性》，载于 *IEEE*国际声学、语音与信号处理会议（*ICASSP*）。IEEE，2020 年，第 7694–7698 页。[14] Cheng-I Lai 等人，《通过自监督语音与语言模型预训练实现半监督口语语言理解》，载于*ICASSP*。IEEE，2021 年。[15] Shu-wen Yang 等人，《SUPERB：语音处理通用性能基准》，*arXiv* 预印本 *arXiv:2105.01051*，2021 年。[16] Neil Houlsby 等人，《面向自然语言处理的参数高效迁移学习》，载于 国际机器学习会议，2019 年，第 2790–2799 页。[17] Yun Zhu 等人，《采用自注意力结构化参数化的多语种自动语音识别》，载于 *INTERSPEECH*，2020 年，第 4741–4745 页。[18] Carlos Escolano、Marta R Costa-juss`a、Jos´e AR Fonol- losa 和 Carlos Segura，《通过特定语言编码器和解码器实现零样本多语种口语语言翻译》，载于*ASRU*。IEEE，2021 年。[19] Samuel Kessler、Bethan Thomas 和 Salah Karout，《一种用于高效且可扩展自监督语音表征学习的基于适配器的预训练方法》，载于 *IEEEICASSP*。IEEE，2022 年，第 3179–3183 页。

[20] Kai-Wei Chang 等人，《生成式口语语言模型在语音处理任务中的提示微调探索》，*arXiv* 预印本 *arXiv:2203.16773*，2022。[21] Kushal Lakhotia 等人，《基于原始音频的生成式口语语言建模》，计算语言学协会汇刊，第 9 卷，2021。[22]Jonas Pfeiffer、Aishwarya Kamath、Andreas R¨uckl´e、Kyunghyun Cho 和 Iryna Gurevych，《AdapterFusion：用于迁移学习的非破坏性任务组合》，*arXiv* 预印本 *arXiv:2005.00247*，2020。[23] Franc。ois Chollet，《Xception：采用深度可分离卷积的深度学习》，载于*IEEE*计算机视觉与模式识别会议论文集，2017 年，第 1251–1258 页。[24] Jie Hu、Li Shen、Samuel Albanie、Gang Sun 和 Enhua Wu，《挤压与激励网络》，2017。[25]Loren Lugosch、Mirco Ravanelli、Patrick Ignoto、Vikrant Singh Tomar 和 Yoshua Bengio，《用于端到端口语语言理解的语音模型预训练》，*arXiv* 预印本 *arXiv:1904.03670*，2019。[26] Soujanya Poria 等人，《MELD：用于对话情感识别的多模态多方数据集》，载于计算语言学协会第 *57* 届年会论文集，2019 年，第 527–536 页。[27] Christophe Veaux 等人，《CSTR VCTK 语料库：用于 CSTR 语音克隆工具包的英语多说话人语料库》，爱丁堡大学语音技术研究中心（*CSTR*），2017。[28] Alexis Conneau 等人，《FLEURS：语音通用表征的少样本学习评测》，*arXiv* 预印本 *arXiv:2205.12446*，2022。[29] Vassil Panayotov、Guoguo Chen、Daniel Povey 和 Sanjeev Khudanpur，《LibriSpeech：基于公版有声读物的自动语音识别语料库》，载于*2015 IEEE*国际声学、语音与信号处理会议（*ICASSP*）。IEEE，2015 年，第 5206–5210 页。[30] Alice Coucke、Alaa Saade、Adrien Ball、Th´eodore Bluche、Alexandre Caulier、David Leroy、Cl´ement Doumouro、Thibault Gisselbrecht、Francesco Caltagirone、Thibaut Lavril 等人，《Snips 语音平台：面向隐私设计语音界面的嵌入式口语语言理解系统》，*arXiv* 预印本 *arXiv:1805.10190*，2018。[31] Guanlong Zhao、Sinem Sonsaat、Alif Silpachai、Ivana Lucic、Evgeny Chukharev-Hudilainen、John Levis 和 Ricardo Gutierrez-Osuna，《L2-Arctic：非母语英语语音语料库》，载于*Interspeech*，2018 年，第 2783–2787 页。

[32]Pete Warden，《Speech Commands：一个用于有限词汇语音识别的数据集》，*arXiv* 预印本 *arXiv:1804.03209*，2018。

[33]Jonathan Shen, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan 等，“通过以梅尔频谱图预测为条件的 WaveNet 实现自然语音合成”，载于*2018*年 *IEEE* 国际声学、语音与信号处理会议（*ICASSP*）。IEEE，2018年，第4779–4783页。

[34] Yi Ren、Yangjun Ruan、Xu Tan、Tao Qin、Sheng Zhao、Zhou Zhao 和 Tie-Yan Liu，“FastSpeech：快速、稳健且可控的文本转语音”，神经信息处理系统进展，第32卷，2019年。

[35] Yi Ren、Chenxu Hu、Xu Tan、Tao Qin、Sheng Zhao、Zhou Zhao 和 Tie-Yan Liu，“FastSpeech 2：快速、高质量的端到端文本转语音”，*arXiv* 预印本 *arXiv:2006.04558*，2020年。[36] Naihan Li、Shujie Liu、Yanqing Liu、Sheng Zhao 和 Ming Liu，“基于 Transformer 网络的神经语音合成”，载于*AAAI*人工智能会议论文集，2019年，第33卷，第6706–6713页。[37] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Lukasz Kaiser 和 Illia Polosukhin，“Attention Is All You Need”，神经信息处理系统进展，第30卷，2017年。[38] Robert Kubichek，“用于客观语音质量评估的梅尔倒谱距离度量”，载于*IEEE* 太平洋地区通信、计算机与信号处理会议论文集。IEEE，1993年，第1卷，第125–128页。[39] “Silero模型：预训练的企业级语音转文本/文本转语音模型及基准测试”，“访问日期：2022-07-10，线上网址：https://github. com/snakers4/silero-models/wiki/Quality-Benchmarks#en-v6。[40]Chao Li、Xiaokong Ma、Bing Jiang、Xiangang Li、Xuewei Zhang、Xiao Liu、Ying Cao、Ajay Kannan 和 Zhenyao Zhu，“Deep Speaker：端到端神经说话人嵌入系统”，*arXiv* 预印本 *arXiv:1705.02304*，2017年。[41] Jungil Kong、Jaehyeon Kim 和 Jaekyoung Bae，“HiFi-GAN：用于高效、高保真语音合成的生成对抗网络”，神经信息处理系统进展，第33卷，第17022–17033页，2020年。

[42] Jonas Pfeiffer、AndreasR¨uckl´e、Clifton Poth、Aishwarya Kamath、Ivan Vuli´c、Sebastian Ruder、Kyunghyun Cho 和 Iryna Gurevych，“Adapterhub：一种用于适配 Transformer 模型的框架”，载于*2020*年自然语言处理实证方法会议论文集（*EMNLP 2020*）：系统演示，线上，2020年，第 46–54 页，计算语言学协会。

7. 附录

7.1. 预训练模型：Wav2Vec2.0

Wav2Vec 2.0 由三个主要模块构成：特征编码器、量化模块和上下文网络。特征编码器由七个时序卷积块组成。它将语音输入 X （16 kHz 的原始单声道波形）转换为潜在表示 z。量化模块将 z 离散化为 q，即由离散语音表征和潜在语音表征组成的字典。最后，由 12 个 Transformer 块构成的上下文网络会生成上下文化嵌入 c，其维度与 q 相同。该模型采用自监督范式进行训练，并优化两个损失函数。第一个损失是对比损失，要求模型利用 c，从输入样本中抽取的有限量化表示集合中，预测某个被遮蔽输入的量化表示 q。第二个损失则确保量化表示具有多样性 [10]。

7.2. TTS架构

本文中 TTS 的基础模型基于 [36]。整体架构如图 6 所示。有关更多详细信息，读者可参考 [36]。在本研究中，TTS 模型训练期间使用的说话人表示通过 [40]中所述的算法计算得到。说话人嵌入的维度为 512。输出的梅尔频谱图通过基于预训练 HiFiGAN [41]的声码器转换为音频。训练和预处理 TTS 所需的其他超参数列于表 6 和表7中。

音频		STFT		MEL	
顶部裁剪分贝	23	滤波器长度	1024	梅尔通道数 n	80
采样率	22050	跳跃长度	256	梅尔最低频率	0
WAV 最大值	32768.0	窗口长度	1024	梅尔最高频率	8000

表 6。将原始音频数据转换为梅尔频谱图所需的音频预处理参数。

优化器		模型	
批量大小	16	编码器层	4
贝塔参数	[0.9, 0.98]	编码器头/解码器头	2/2
梯度裁剪阈值	1.0	编码器隐藏维度/解码器隐藏维度	256/256
总步数	900000	解码器层	6
预热步数	4000	卷积滤波器大小	1024
退火步数	[300000, 400000, 500000]	卷积核大小	[9,1]
退火率	0.3	编码器丢弃率/解码器丢弃率	0.2/0/2

表7。训练和推理期间使用的优化器和模型参数。

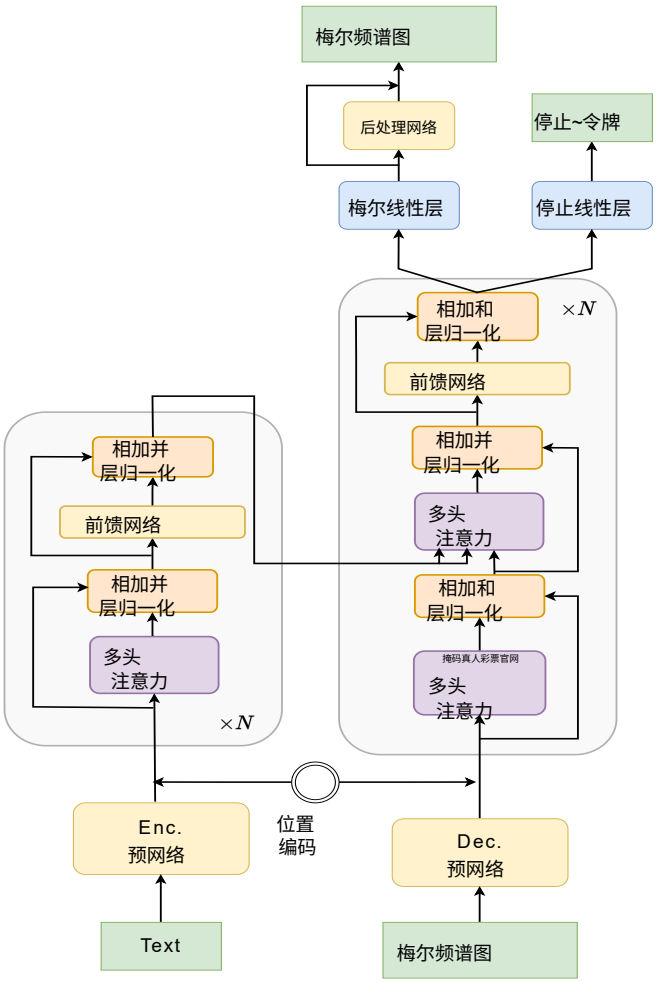


图6。基于 Transformer 的文本转语音框架概览。每个 Transformer 层中均按照图 2 添加适配器。

7.3. 超参数

本节介绍本文中不同任务所用的超参数，以生成实验结果。我们针对每项任务尝试了不同的学习率，并报告各算法取得最佳结果时的取值。我们没有探索前缀微调和 LoRA 的超参数，而是采用 [42]中提到的默认配置。未来的工作可以探索这些适配器的不同配置。表 8 列出了不同语音处理任务的所有参数。

任务/方法	数据集	微调		适配器			前缀		LoRA			ConvAdapter		
		lr	批量大小	lr	批量大小	c	lr	批量大小	lr	批量大小	r	lr	批量大小	c
SR	ESD	2e-4	64	2e-2	64	2	2e-2	64	2e-4	64	8	2e-4	64	2
	VCTK	8e-5	64	2e-5	64	2	2e-3	64	2e-4	64	8	2e-4	64	2
SER	ESD	2e-4	64	2e-2	64	2	2e-2	64	2e-4	64	8	2e-3	64	2
	MELD	7e-6	64	8e-6	64	2	2e-2	64	8e-4	64	8	2e-4	64	2
ASR	ESD	8e-6	16	2e-5	16	2	2e-2	16	2e-3	16	8	2e-4	16	2
	FLEURS	1e-4	32	1e-4	32	2	2e-3	32	2e-3	32	8	5e-3	32	2
	LibriSpeech	3e-5	64	1e-4	64	2	3e-4	64	3e-4	64	8	3e-4	64	2
PR	LibriSpeech	3e-4	16	3e-4	16	2	3e-4	16	2e-2	16	8	2e-4	16	2
KS	Speech Commands	8e-6	64	8e-6	64	2	2e-4	64	8e-6	64	8	2e-4	64	2
IC	流畅命令	2e-4	8	2e-4	8	2	2e-4	8	2e-4	8	8	2e-4	8	2
SF	SNIPS	2e-4	8	2e-4	8	2	2e-3	8	2e-3	8	8	2e-4	8	2
TTS	LibriTTS	1.7e-6	16	1.7e-6	16	2	1.7e-6	16	1.7e-6	16	8	1.7e-6	16	2
	L2Arctic	1.7e-6	16	1.7e-6	16	2	1.7e-6	16	1.7e-6	16	8	1.7e-6	16	2

表 8。训练期间不同任务所用超参数的详细信息。其中，lr、c 和 r 分别表示学习率、压缩率和秩。