arXiv:2301.02966v1 [cs.CL] 8 Jan 2023

SPEECHAIN：A 面向大规模机器语音链的语音工具包

Heli Qi[1], *Sashi Novitasari*[1*], *Andros Tjandra*[1†], *Sakriani Sakti*[2],*Satoshi Nakamura*[1]

[1]日本奈良先端科学技术大学院大学[2]日本先进科学技术大学

摘要

本文介绍了 SpeeChain，这是一个基于 PyTorch 构建的开源工具包，旨在开发可大规模应用的机器语音链。本次首次发布聚焦于文本转语音→自动语音识别链，这是机器语音链的核心组成部分，指的是利用未朗读文本进行文本转语音数据增强，以提升自动语音识别。为构建高效的大规模文本转语音→自动语音识别链处理流程，我们实现了易于使用的多GPU批次级模型推理、多数据加载器批次生成以及动态数据选择技术。本文首先说明文本转语音→自动语音识别链的整体流程及各步骤面临的困难。随后，我们针对不同类型的无标签数据、数据过滤阈值、批次组成以及真实与合成数据比例，开展了详细的消融研究。在 LibriSpeech 的训练集 *clean 460*上进行的实验结果表明，在半监督设置下，我们的文本转语音→自动语音识别链能够显著降低词错误率。‡

索引词— 开源语音处理工具包、机器语音链、自动语音识别、文本转语音、半监督学习

1. 引言

端到端（E2E）自动语音识别（ASR） [1–4]和文本转语音合成（TTS） [5–7]受益于深度学习的最新进展，取得了巨大成功。自动语音识别和文本转语音被认为彼此对称，因为它们共享同类型的训练数据，只是输入和输出互换。尽管两者关系密切，但在 [8, 9] 提出机器语音链之前，这两个领域的研究基本上各自独立发展。自 [10, 11]以来，端到端自动语音识别-文本转语音的研究受到了更多关注。

机器语音链是一种基于深度学习的闭环架构，将端到端自动语音识别和文本转语音模型连接起来。机器语音链的一项核心应用是文本转语音→自动语音识别链，其中，端到端自动语音识别模型使用无标签文本，由端到端文本转语音模型生成合成语音进行训练。文本转语音→自动语音识别链在单说话人 [8] 和多说话人数据集 [12] 上的初步实验取得了成功，因此人们开始关注利用多说话人文本转语音模型生成的合成语音来训练端到端自动语音识别模型。社区中已有研究 [13–17]使用多说话人文本转语音模型生成的合成语音。对于多说话人文本转语音，研究者引入了说话人嵌入模型，帮助文本转语音系统捕捉不同的说话人身份，并生成具有不同说话风格的合成语音。此外，[16] 提出了另外两种嵌入编码器，用于提取局部和全局话语嵌入，以提高文本转语音保真度。除了说话人多样性之外，[16–19] 还为文本转语音系统提供了无标签域外文本，帮助自动语音识别模型克服语言不匹配。

在语音处理社区中，开发了许多优秀的开源工具包 [20–25]，用于支持自动语音识别和文本转语音模型，并且其性能已达到与最先进水平相当的程度。然而，这些一体化工具包大多独立开发自动语音识别和文本转语音组件，二者之间几乎没有关联。为弥合这一差距并建立自动语音识别与文本转语音之间的联系，我们推出了 SpeeChain。这是一款工具包，为研究人员和开发者提供了高效的文本转语音→自动语音识别处理流程。SpeeChain 支持多种模块化功能，例如利用多个 GPU 快速进行模型推理、为真实数据和合成数据提供统一存储结构，以及实时数据筛选与加载。此外，我们还为自动语音识别和文本转语音模型提供了经过深度调优的超参数，以促进自动语音识别与文本转语音联合建模的研究。

本文其余部分的组织结构如下：第 2 节介绍文本转语音→自动语音识别处理流程。第 3 节解释我们的工具包针对文本转语音→自动语音识别处理流程所采用的技术方案。第 4 节展示我们的实验结果。在第 5 节中，我们总结本文工作并介绍未来的研究计划。

2. 面向自动语音识别的文本转语音数据增强（文本转语音→自动语音识别链）

文本转语音→自动语音识别链的流程如图 1 所示。

2.1. 基础自动语音识别与文本转语音训练

起初，基础自动语音识别和文本转语音模型在同一个有标签数据集上进行训练。基础模型的性能在很大程度上决定了下一阶段合成语音的质量及其词错误率评估结果。然而，由于有限有标签数据的分布不完整，很难训练出

*Sashi Novitasari 目前任职于 IBM东京研究实验室。
†Andros Tjandra 目前任职于 Meta AI。
‡我们将在论文发表后发布工具包。

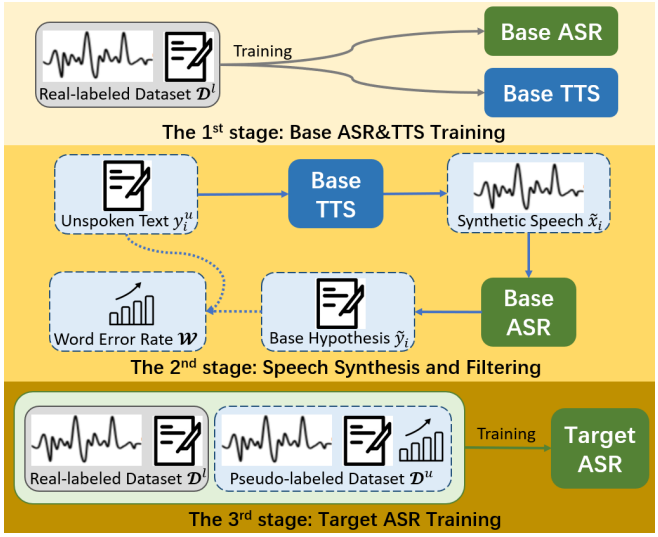


图 1。文本转语音→自动语音识别链概览。

性能足够良好的基础模型，尤其是多说话人文本转语音模型。常用策略包括延长预热阶段、降低学习率、提高丢弃率 [26], 缩减模型参数等。

2.2. 语音合成与过滤

在此阶段，给定无标签数据集 $\mathcal{D}^u = \{y_1^u, \ldots, y_{|\mathcal{D}^u|}^u\}$，基础文本转语音模型 θ_{TTS}^{base}会为未朗读文本 y_i^u生成合成语音 $\tilde{x}_i$（$i \in \{1, \ldots, |\mathcal{D}^u|\}$），其形式如下：

$$\tilde{x}_i = \arg\max_z \prod_{t=1}^{T} p(z_t | \tilde{x}_{i,0:t-1}, y_i^u; \theta_{TTS}^{base}). \quad (1)$$

合成语音 $\tilde{x}_i$ 与未朗读文本y_i^u组合，形成伪标注数据集 $\mathcal{D}^u = \{(\tilde{x}_1, y_1^u) \ldots\ldots, (\tilde{x}_{|\mathcal{D}^u|}, y_{|\mathcal{D}^u|}^u)\}$。然而，尽管在训练基础文本转语音模型上投入了大量精力，由于可用有标签数据量有限，合成语音中仍可能存在一些错误（例如发音错误、静音、重复短语）。通常会在最终阶段之前进行数据过滤，以减少合成语音中的错误。文本转语音→自动语音识别链使我们能够通过基础词错误率（WER） $\mathcal{W}^{base}$ 来过滤合成语音，其形式如下：

$$\tilde{y}_i = \arg\max_z \prod_{t=1}^{T} p(z_t | \tilde{y}_{i,0:t-1}, \tilde{x}_i; \theta_{ASR}^{base}), \quad (2)$$

$$\mathcal{W}_i^{base} = \frac{edit_distance(\tilde{y}_i, y_i^u)}{|y_i^u|} \quad (3)$$

其中，$\tilde{y}_i$ 是基础自动语音识别模型对$\tilde{x}_i$生成的假设转写，$edit\ distance()$ 是用于衡量 $\tilde{y}_i$ 和 y_i^u的词序列之间最小编辑距离的函数，$|y_i^u|$ 是 y_i^u中的词数。

2.3. 目标自动语音识别训练

在最终阶段，$\mathcal{W}^{base}$ 大于给定阈值的合成语音将首先从伪标注数据集 $\mathcal{D}^u$ 中筛除。然后，将真实标注数据集 $\mathcal{D}^l = \{(x_1^l, y_1^l) \ldots, (x_{|\mathcal{D}^l|}^l, y_{|\mathcal{D}^l|}^l)\}$ 与 $\mathcal{D}^u$合并，得到扩充后的数据集 $\mathcal{D} = \mathcal{D}^l \cup \mathcal{D}^u$，并用它训练目标 ASR 模型 θ_{ASR}^{target}。训练时，单个批次 $\mathcal{B} = \mathcal{B}^l \cup \mathcal{B}^u$ 由按比例从 $\mathcal{D}^l$ 和 $\mathcal{D}^u$中抽取的真实标注数据$\mathcal{B}^l = \{(x_1^l, y_1^l) \ldots, (x_{|\mathcal{B}^l|}^l, y_{|\mathcal{B}^l|}^l)\}$和伪标注数据$\mathcal{B}^u = \{(\tilde{x}_1, y_1^u) \ldots, (\tilde{x}_{|\mathcal{B}^u|}, y_{|\mathcal{B}^u|}^u)\}$组成。分别在 $\mathcal{B}^l$ 和 $\mathcal{B}^u$ 上计算两个训练损失 $\mathcal{L}^l$ 和 $\mathcal{L}^u$，即

$$\mathcal{L}^l = -\frac{1}{|\mathcal{B}^l|} \sum_{i=1}^{|\mathcal{B}^l|} \sum_{t=1}^{T} \log p(y_{i,t}^l | y_{i,0:t-1}^l, x_i^l; \theta_{ASR}^{target}), \quad (4)$$

$$\mathcal{L}^u = -\frac{1}{|\mathcal{B}^u|} \sum_{i=1}^{|\mathcal{B}^u|} \sum_{t=1}^{T} \log p(y_{i,t}^u | y_{i,0:t-1}^u, \tilde{x}_i; \theta_{ASR}^{target}), \quad (5)$$

其中，语音 x_i 和前缀词元 $y_{0:t-1}$ 均作为输入，采用教师强制技术。最后，通过总体损失 $\mathcal{L} = (1-\lambda)\mathcal{L}^l + \lambda\mathcal{L}^u$ 对目标 ASR 模型进行优化，该损失由手动设置的权重 λ控制。

3. 工具包特性

与许多工具包一样，基础模型训练遵循常规的监督训练方案，而最后两个阶段则需要更有针对性的设计，以提高处理流程的效率。我们的工具包实现了以下独特功能，以简化文本转语音→自动语音识别链的处理流程。

3.1. 采用统一数据结构的快速模型推理 构

由于端到端ASR和TTS模型具有自回归特性，第二阶段对大量无标签数据进行模型推理会耗费大量时间。我们的工具包通过 *torch.nn.parallel.Distributed-DataParallel* 实现批次级模型推理，同时利用多个 GPU 加速模型推理。在推理过程中，我们将无标签数据划分为多个互不重叠的片段。每个 GPU 都运行一个独立进程，在一次推理步骤中对其负责部分的一批无标签数据进行解码。所有 GPU 完成各自的任务后，我们会合并全部推理结果，形成最终的伪标注数据集。图2展示了使用多个 GPU 进行文本转语音解码的示意图。

在模型推理过程中，我们将模型计算与数据存储解耦，使模型假设和推理元数据（例如，词错误率、ASR置信度）能够与真实数据集采用相同的结构。真实数据集与伪数据集使用统一的数据结构，大幅减少了开展实验所需的工作量。

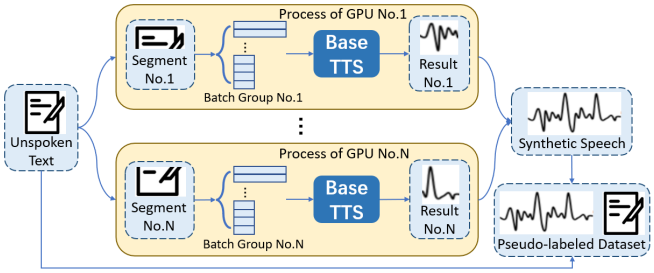


图 2。多个 GPU 执行批次级 TTS 推理的流程图。我们按照无标签文本的长度总和对其分组，以确保每个批次中的合成帧数量相同。自动语音识别的波束搜索采用了类似的机制。

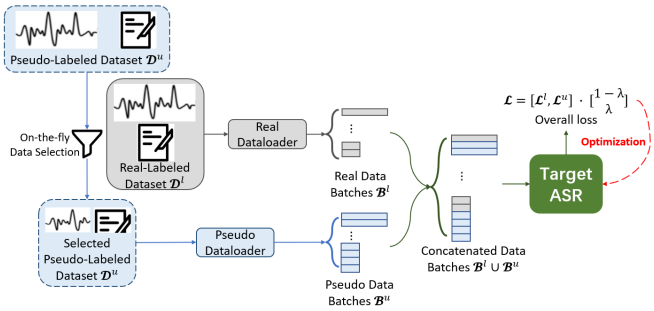


图 3。使用两个数据加载器并进行动态数据选择的数据加载流程图。我们将话语长度相近的真实数据批次和伪数据批次拼接起来，以减少模型前向传播操作的次数。

3.2. 使用多个数据加载器进行批次生成

在最终阶段训练目标自动语音识别模型时，将真实标注数据集与伪标注数据集混合，并从混合数据集中动态提取数据，并不是一个好主意。在大多数情况下，无标签数据远多于有标签数据，因此许多数据批次都会由伪标注数据主导。这样一来，就很难控制每个数据批次的梯度方向，训练也会变得不稳定。

我们的工具包使用多个 *torch.utils.data.DataLoader*，分别从不同数据集中获取语音文本对，并生成真实数据与伪数据比例固定的批次，如图 3 所示。固定的数据比例不仅能够规范每个批次计算出的梯度方向，还能清晰呈现每个批次中真实数据与伪数据的构成。清晰的构成使我们能够设计更先进的算法，以便在训练过程中更好地利用伪标注数据。

3.3. 伪标注数据的在线数据选择

我们的工具包将批次生成与数据集访问（即 *torch.utils.data.Dataset*）解耦，因此我们可以轻松更改每个数据加载器可访问的语音文本对

工作（模型）	dev_clean	test_clean	test_other
LibriSpeech 基线——*train clean 100*			
[16] (LAS [2])	-	12.46%	34.00%
[27] (TDS [4])	14.00%	14.85%	39.95%
[28](Transformer [3])	12.20%	12.90%	31.10%
我们的方法 (Transformer)	**11.90%**	**12.20%**	**30.19%**
上限为 LibriSpeech-训练集 *clean 100* 和 LibriSpeech-训练集 *clean 360*			
[16] (LAS)	-	6.30%	22.41%
[18] (LAS)	-	6.50%	-
我们的方法（Transformer）	5.08%	**5.65%**	**15.91%**
上限基于 LibriSpeech-训练集 *clean 100* &LibriTTS-训练集 *clean 360*			
我们的方法（Transformer）	6.14%	6.57%	19.50%

表1。与文献中的结果相比，未使用语言模型时自动语音识别基线和上限的词错误率（WER）。基线采用包含 1k 个 BPE词元的词汇表；上限采用包含 5k 个 BPE词元的词汇表。我们在 LibriSpeech 和 LibriTTS 上训练了两个顶线模型，以便与我们的文本转语音→自动语音识别链进行公平比较。

以便在训练期间使用给定的元数据。我们还提供元数据分布直方图，帮助用户选择适当的阈值，以过滤掉低质量的伪标注数据。

4. 实验

4.1. 监督式学习的 ASR基线和上限

我们的自动语音识别模型基于 Speech-Transformer [3], , 其中编码器和解码器分别由 12 层和 6 层 Transformer 组成。我们为自动语音识别基线（d_{model}=256， n_{head}=4）和上限（d_{model}=512， n_{head}=8）设置了两种模型配置。这两种配置采用相同的 2048 维前馈层。在所有实验中，我们统一使用 16 个波束进行自动语音识别解码，且不使用 CTC分支。基线模型和上限模型的结果见表1。

4.2. 基础文本转语音模型训练

如 [15], 所述， [29] LibriSpeech [30]不适合用于文本转语音模型训练，因此我们将 LibriTTS 下采样至 16kHz，用于基础文本转语音训练（*trainclean 100*）和语音合成（*train clean 360*）。我们的文本转语音模型基于 MultiSpeech [7], , 其编码器和解码器均由相同的 6 个 Transformer 层组成（d_{model}=512、n_{head}=8、 d_{ff}=2048）。编码器直接以字符作为输入。

我们观察到，文本转语音比自动语音识别更需要大量数据。因此，我们将所有小写字母转换为大写形式，并移除逗号、句号和单引号以外的标点符号，以帮助文本转语音收敛。我们让基础

过滤阈值	dev_clean	test_clean	test_other
LibriTTS 的文本转语音→自动语音识别链：训练集 *clean_360*			
基准词错误率<10%	9.27%	10.22%	27.12%
基础词错误率<20%	8.48%	9.30%	25.90%
基础词错误率<30%	8.11%	8.68%	24.82%
基础词错误率<50%	7.83%	8.76%	25.65%
基础词错误率<100%	7.81%	8.45%	24.50%
不进行基线词错误率筛选	**7.32%**	**8.07%**	**24.17%**
基于 LibriSpeech-*train clean 360* 的自动语音识别自训练			
置信度最高的20%	10.23%	10.80%	26.28%
置信度最高的40%	9.04%	9.75%	24.26%
置信度最高的60%	8.86%	9.47%	23.44%
置信度排名前 80%	**8.51%**	9.36%	22.76%
不进行置信度筛选	8.67%	**9.33%**	**22.66%**

表 2。不同过滤阈值下目标 ASR 模型的词错误率。

批次组成	dev_clean	test_clean	test_other
上限为 LibriSpeech-训练 *clean 100*& LibriTTS-训练 *clean 360*			
静态 R2S 比率	6.37%	6.82%	19.75%
动态 R2S 比率	**6.14%**	**6.57%**	**19.50%**
10% 基础 WER 筛选（2:1）			
静态 R2S 比率	**9.27%**	**10.22%**	**27.12%**
动态 R2S 比率	10.59%	11.39%	28.97%
不进行基线词错误率筛选 (1:2)			
静态 R2S 比率	**7.32%**	**8.07%**	**24.17%**
动态 R2S 比率	7.75%	8.54%	24.93%

表3。不同批次组成下目标 ASR 模型的词错误率。R2S 比例表示每个训练批次中真实数据与合成数据的比例。括号中的数字表示整个训练集的真实数据与合成数据比例。

文本转语音模型在每个解码步骤生成四个连续帧，以加快语音合成。我们不采用神经声码器，而是对自动语音识别和文本转语音训练使用相同的 80 维对数梅尔频谱特征，这些特征通过 50 毫秒的窗长和 10 毫秒的窗移提取。此外，我们发现，在添加到位置编码之前，将编码器和解码器嵌入向量按 $\sqrt{d_{model}}$ 进行缩放，有助于基础文本转语音训练。在文本转语音语音合成过程中，我们从 LibriTTS-*train clean 100* 中随机抽取说话人，作为基础文本转语音模型的参考说话人。

4.3. 通过基础自动语音识别模型进行词错误率筛选

我们的目标 ASR 模型与顶线模型采用相同的配置。我们按照表 2 所示改变基础词错误率筛选阈值，其中 λ 统一设为 0.5。我们将文本转语音→自动语音识别链与自动语音识别自训练 [27],进行比较；在后者中，基础自动语音识别模型为未转录语音生成伪文本，以此自行扩充训练集。在自动语音识别自训练实验中，我们使用 LibriSpeech-*train clean 360* 作为无标签数据，并将自动语音识别解码置信度用作筛选标准。

结果表明，基础词错误率筛选反而降低了自动语音识别的性能提升。我们发现，由于真实语音与合成语音之间存在不匹配，基础自动语音识别模型给出的词错误率无法评估合成语音的质量。然而，即使合成语音中存在一些错误，目标自动语音识别模型仍能从基于合成语音的训练中受益。我们推测，合成语音中的某些错误可能起到数据增强的作用，例如，静音可能会促使自动语音识别模型预测缺失部分。

我们还观察到，文本转语音→自动语音识别链在*test other* 上的提升小于自动语音识别自训练。我们推测，*clean* 与 *other* 数据集中的语音数据存在声学失配。由于我们的文本转语音模型仅使用*clean* 数据进行训练，合成语音无法像真实未转录语音那样，为目标 ASR 模型提供丰富的声学细节。

提供声学细节。

4.4 真实数据与合成数据组成的影响

为研究第3.2节中提到的批次组成对自动语音识别训练的影响，我们针对表3所示的不同批次生成方式进行了对比实验。我们考虑了三种情形，训练集分别包含 100%、66% 和 33% 的真实语音。在每种情形下，批次组成要么由两个数据加载器静态确定，要么由一个数据加载器动态确定。结果表明，只要训练集中含有合成语音，就有必要在每个批次中加入一定量的真实语音，以规范梯度方向。但对于使用两个真实数据集进行的上限训练，最好将它们混合起来，使模型训练在批次生成过程中获得更多随机性。

5. 结论与未来工作

本文介绍了 SpeeChain，这是一款专为大规模机器语音链设计的新型工具包。首个版本提供了一套易于使用的文本转语音数据增强处理流程，用于半监督自动语音识别。我们的大量实验表明，通过文本转语音→自动语音识别链精心开发的半监督自动语音识别模型，其性能可接近使用真实标注数据训练的监督式模型。未来，我们将实现各种先进的自动语音识别和文本转语音模型，并开发机器语音链的更多应用，例如用于半监督文本转语音的自动语音识别→文本转语音链，以及用于自动语音识别-文本转语音联合优化的在线语音链学习。

6. 致谢

本研究部分由 JSPS KAKENHI（资助编号 JP21H05054 和 JP21H03467）和 Google AI 资助。

7. 参考文献

[1] Jan K Chorowski、Dzmitry Bahdanau、Dmitriy Serdyuk、Kyunghyun Cho 和 Yoshua Bengio，《用于语音识别的基于注意力的模型》，神经信息处理系统进展，第 28 卷，2015 年。[2] William Chan、Navdeep Jaitly、Quoc Le 和 Oriol Vinyals，《听、注意与拼写：用于大词汇量会话语音识别的神经网络》，载于 *2016 IEEE*声学、语音与信号处理国际会议（*ICASSP*）。IEEE，2016 年，第 4960–4964 页。[3] Linhao Dong、Shuang Xu 和 Bo Xu，《Speech-Transformer：用于语音识别的无循环序列到序列模型》，载于 *2018 IEEE*声学、语音与信号处理国际会议（*ICASSP*）。IEEE，2018 年，第 5884–5888 页。[4] Awni Hannun、Ann Lee、Qiantong Xu 和 Ronan Collobert，《采用时间深度可分离卷积的序列到序列语音识别》，*arXiv* 预印本 *arXiv:1904.02619*，2019 年。[5] Jonathan Shen、Ruoming Pang、Ron J Weiss、Mike Schuster、Navdeep Jaitly、Zongheng Yang、Zhifeng Chen、Yu Zhang、Yuxuan Wang、Rj Skerrv-Ryan 等，《通过基于梅尔频谱图预测对 WaveNet 进行条件控制的自然 TTS 合成》，载于 *2018 IEEE*声学、语音与信号处理国际会议（*ICASSP*）。IEEE，2018 年，第 4779–4783 页。[6] Naihan Li、Shujie Liu、Yanqing Liu、Sheng Zhao 和 Ming Liu，《采用 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 33-01 卷，第 6706–6713 页。[7] Mingjian Chen、Xu Tan、Yi Ren、Jin Xu、Hao Sun、Sheng Zhao、Tao Qin 和 Tie-Yan Liu，《MultiSpeech：采用 Transformer 的多说话人文本转语音》，*arXiv* 预印本 *arXiv:2006.04664*，2020 年。[8] Andros Tjandra、Sakriani Sakti 和 Satoshi Nakamura，《边说边听：基于深度学习的语音链》，载于 *2017 IEEE*自动语音识别与理解研讨会（*ASRU*）。IEEE，2017 年，第 301–308 页。[9] Andros Tjandra、Sakriani Sakti 和 Satoshi Nakamura，《机器语音链》，*IEEE/ACM* 音频、语音与语言处理汇刊，第 28 卷，第 976–989 页，2020 年。

[10] Yi Ren、Xu Tan、Tao Qin、Zhou Zhao、Zhao Zhou 和 Tie-Yan Liu，“Yan Liu、Tao Qin监督的文本转语音和无监督的语音识别”，载于国际机器学习大会。PMLR，2019 年，第 5410–5419 页。

[11] Gary Wang、Andrew Rosenberg、Zhehuai Chen、Yu Zhang、Bhuvana Ramabhadran、Yonghui Wu 和 Pedro Moreno，“利用合成语音上的一致预测改进语音识别”，载于*2020*年*IEEE*国际声学、语音与信号处理会议*(ICASSP)*。IEEE，2020 年，第 7029–7033 页。

[12] Andros Tjandra、Sakriani Sakti 和 Satoshi Nakamura，“带单次说话人自适应的机器语音链”，*arXiv* 预印本 *arXiv:1803.10525*，2018 年。

[13] Nick Rossenbach、Mohammad Zeineldeen、Benedikt Hilmes、Ralf Schl¨uter 和 Hermann Ney，“比较合成训练数据对各种自动语音识别架构的益处”，载于*2021 IEEE* 自动语音识别与理解研讨会（*ASRU*）。IEEE，2021 年，第 788–795 页。

[14] Jason Li、Ravi Gadde、Boris Ginsburg 和 Vitaly Lavrukhin，《使用合成语音增强训练神经语音识别系统》，*arXiv* 预印本 *arXiv:1811.00707*，2018 年。

[15] Nick Rossenbach、Albert Zeyer、Ralf Schl¨uter 和 Hermann Ney，《为基于注意力的语音识别系统生成合成音频数据》，载于*2020*年*IEEE*国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2020年，第7069–7073页。

[16] Andrew Rosenberg、Yu Zhang、Bhuvana Ramabhadran、Ye Jia、Pedro Moreno、Yonghui Wu 和 Zelin Wu，《利用增强的合成语音进行语音识别》，载于*2019 IEEE* 自动语音识别与理解研讨会 *(ASRU)*。IEEE，2019年，第996–1002页。[17] Sei Ueno、Masato Mimura、Shinsuke Sakai 和 Tatsuya Kawahara，《用于声学到词语音识别数据增强的多说话人序列到序列语音合成》，载于*2019 IEEE*国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2019年，第6161–6165页。[18] Fengpeng Yue、Yan Deng、Lei He 和 Tom Ko，《探索用于领域自适应和少样本说话人自适应的机器语音链》，*arXiv*预印本 *arXiv:2104.03815*，2021年。[19] Murali Karthick Baskar、Luk´aˇs Burget、Shinji Watanabe、Ramon Fernandez Astudillo 等，《EAT：用于自监督语音识别的增强型自动语音识别-文本转语音》，载于*2021 IEEE*国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2021年，第6753–6757页。[20] Shinji Watanabe、Takaaki Hori、Shigeki Karita、Tomoki Hayashi、Jiro Nishitoba、Yuya Unno、Nelson Enrique Yalta Soplin、Jahn Heymann、Matthew Wiesner、Nanxin Chen 等，《ESPnet：端到端语音处理工具包》，*arXiv*预印本 *arXiv:1804.00015*，2018年。[21]Tomoki Hayashi、Ryuichi Yamamoto、Katsuki Inoue、Takenori Yoshimura、Shinji Watanabe、Tomoki Toda、Kazuya Takeda、Yu Zhang 和 Xu Tan，《ESPnet-TTS：统一、可复现且可集成的开源端到端文本转语音工具包》，载于*2020*年*IEEE*国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2020年，第7654–7658页。[22] Myle Ott、Sergey Edunov、Alexei Baevski、Angela Fan、Sam Gross、Nathan Ng、David Grangier 和 Michael Auli，《fairseq：一种快速、可扩展的序列建模工具包》，*arXiv*预印本 *arXiv:1904 01038*，2019年。。[23] Changhan Wang、Wei-Ning Hsu、Yossi Adi、Adam Polyak、Ann Lee、Peng-Jen Chen、Jiatao Gu 和 Juan Pino，《fairseq sˆ2：一种可扩展且可集成的语音合成工具包》，*arXiv*预印本 *arXiv:2109 06912*，2021年。。[24] Mirco Ravanelli、Titouan Parcollet、Peter Plantinga、Aku Rouhe、Samuele Cornell、Loren Lugosch、Cem Subakan、Nauman Dawalatabad、Abdelwahab Heba、Jianyuan Zhong 等，《SpeechBrain：通用语音工具包》，*arXiv*预印本 *arXiv:2106 04624*，2021年。。[25] Hui Zhang、Tian Yuan、Junkun Chen、Xintong Li、Renjie Zheng、Yuxin Huang、Xiaojie Chen、Enlei Gong、Zeyu Chen、Xiaoguang Hu 等，《PaddleSpeech：易于使用的一体化语音工具包》，*arXiv*预印本 *arXiv:2205.12007*，2022年。[26] Nitish Srivastava、Geoffrey Hinton、Alex Krizhevsky、Ilya Sutskever 和 Ruslan Salakhutdinov，《Dropout：一种防止神经网络过拟合的简单方法》，《机器学习研究杂志》，第15卷，第1期，第1929–1958页，2014年。[27] Jacob Kahn、Ann Lee 和 Awni Hannun，《端到端语音识别的自训练》，载于*2020*年*IEEE*国际声学、语音与信号处理会议 *(ICASSP)*。IEEE，2020年，第7084–7088页。[28] Yosuke Higuchi、Niko Moritz、Jonathan Le Roux 和 Takaaki Hori，《动量伪标签：利用持续改进的伪标签进行半监督自动语音识别》，*IEEE*《信号处理精选主题期刊》，2022年。[29] Vassil Panayotov、Guoguo Chen、Daniel Povey 和 Sanjeev Khudan- pur，《LibriSpeech：基于公有领域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] Heiga Zen、Viet Dang、Rob Clark、Yu Zhang、Ron J Weiss、Ye Jia、Zhifeng Chen 和 Yonghui Wu，《LibriTTS：从 LibriSpeech 衍生的文本转语音语料库》，*arXiv*预印本 *arXiv:1904.02882*，2019年。