arXiv:2303.03849v3 [eess.AS] 1 Jan 2024

TS-SEP：基于估计的说话人嵌入进行联合说话人日志与分离

Christoph Boeddeker、Aswin Shanmugam Subramanian、Gordon Wichern、Reinhold Haeb-Umbach 和 Jonathan Le Roux

摘要——鉴于会议数据的说话人日志与声源分离是密切相关的任务，本文提出一种同时实现这两个目标的方法。该方法基于目标说话人语音活动检测（TS-VAD）说话人日志方法，该方法假设已有初始说话人嵌入。我们将 TS-VAD 最后的联合说话人活动估计网络替换为一个能够以时频分辨率生成说话人活动估计值的网络。这些估计值可用作掩码，通过掩蔽或波束形成进行声源提取。该技术适用于单通道和多通道输入，并且在这两种情况下，均在 LibriCSS 会议数据识别任务上取得了新的最先进词错误率（WER）。此外，我们还计算了说话人感知和说话人无关的 WER，以单独评估说话人分离错误对整体 WER 表现的影响。

索引词—会议识别、会议分离、说话人日志

I. 引言

CURRENT 会议转录研究不仅关注将会议的音频录音转录为机器可读文本，还关注通过“谁在何时发言”的说话人分离信息来丰富转录内容。这两项任务的大部分难点都可归因于多人交谈语音的互动动态。说话人以间歇性的方式进行表达，其间交替出现语音静默、单一说话人语音和多人语音片段。尤其是重叠语音，即两人或多人同时说话的情况，已知不仅对自动语音识别构成重大挑战，也对说话人日志构成重大挑战 [1]。

对于应先从说话人日志还是从分离/增强模块开始处理，并没有明显的选择[2]。支持先分离重叠语音片段、同时去除其他声学失真，然后再进行说话人日志的一个论据是：在无重叠语音上执行说话人分离任务要容易得多。这一处理顺序在 [2]中得到了倡导。此外，连续语音分离（CSS）流程也是从分离模块开始的 [3]。早期的说话人日志系统，例如在

C. Boeddeker 和 R. Haeb-Umbach 任职于帕德博恩大学，德国帕德博恩 33098（电子邮箱：{boeddeker、haeb}@nt.upb.de）。A. S. Subramanian 曾任职于三菱电机研究实验室（MERL），美国马萨诸塞州剑桥 02139；现任职于微软公司，One Microsoft Way，雷德蒙德，美国华盛顿州 98052（电子邮箱：asubra13@alumni.jh.edu）。G. Wichern 和 J. Le Roux 任职于三菱电机研究实验室（MERL），美国马萨诸塞州剑桥 02139（电子邮箱：{wichern、leroux}@merl.com）。

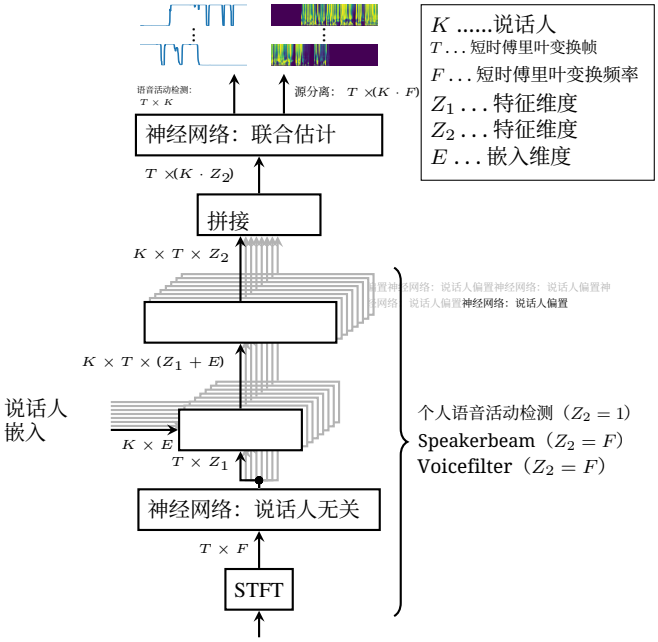


图 1. 单个麦克风上的 TS-VAD 和 TS-SEP 结构。说话人以“阴影”表示。

DIHARD 说话人日志挑战赛的早期几届中占主导地位的基于聚类的方法，通常无法妥善处理重叠语音，因此在具有大量并发语音的数据上表现不佳 [4]。

另一方面，从说话人日志组件开始处理可能更具优势，因为如果已知语音缺失、单一说话人语音和多人语音的片段边界，声源提取就会更加容易。例如，CHiME-6 基线系统 [1] 采用了这种处理顺序。然而，这要求说话人日志组件能够处理重叠语音。

这些不同的观点凸显了这两项任务之间的相互依赖性，也清楚表明，对说话人日志与分离/源提取进行联合处理可能会带来益处。事实上，两者需要解决的子任务相似：说话人日志的任务是确定每个时间帧中处于活动状态的说话人，而基于掩码的源提取本质上是识别每个时频（TF）单元中的活跃说话人，因此二者的区别仅在于分辨率不同，即时间分辨率与时频分辨率的区别。

早期的说话人分离与声源分离联合系统示例包括采用时变混合权重的空间混合模型（SMM）。

在这些模型中，对混合成分先验概率的估计值经过适当量化后，可提供谁在何时说话的说话人分离信息；而后验概率具有时频分辨率，可通过掩蔽或波束形成来提取混合语音中的每个声源。一个关键问题是混合权重或后验概率的初始化。在引导式声源分离（GSS）框架 [7], 中，该框架是 CHiME-6 挑战赛 [1],基线系统的一部分，使用片段边界的人工标注及其后续估计值 [8], 来初始化时变混合权重。文献 [6]提出了一种利用会议数据特点的初始化方案：大多数时间只有一位说话人处于活动状态。

然而，这种联合说话人日志与分离方法依赖于多通道输入的可用性，并且由于期望最大化（EM）算法具有迭代特性，其计算开销很大。此外，至少在其原始形式下，EM算法是一种离线算法，不太适合处理任意时长的会议。EM算法存在递归在线变体，但其计算量仍然很大，并且需要谨慎进行参数调优，尤其是在动态场景中 [9]。

鉴于神经网络近期在说话人日志和声源分离方面均取得成功，它们似乎有望为联合说话人日志与分离/提取问题开发一种集成解决方案。从混合语音中提取目标说话人的知名方法包括 Speakerbeam [10]和 Voicefilter [11]。两者都需要待提取目标说话人的注册话语。在 Speakerbeam 方法中，目标说话人嵌入与提取网络联合训练，而 Voicefilter 则采用单独训练的说话人嵌入计算网络。然而，它们对注册句子的依赖在一定程度上限制了其在某些应用中的实用性。此外，它们是针对典型的声源分离/提取场景设计的，在该场景中，先验已知目标说话人大部分时间处于活跃状态。直到最近，才提出了 Speakerbeam 系统的扩展版本，以处理会议数据 [12]。

在说话人日志领域，端到端神经说话人日志（EEND）近年来取得了重大进展，因为它将说话人日志建模为多标签分类问题，从而能够处理重叠语音 [13]–[16]。EEND以数秒为单位的语音块处理输入语音。这会引入块置换问题，例如可以通过后续聚类来解决 [17]。

目标说话人语音活动检测（TS-VAD）说话人日志系统 [8], [18], 源自早期的个人语音活动检测方法 [19],，在极具挑战性的 CHiME-6 数据集 [1]上展现出优异性能。假设已知会议中的说话人总数，且有表示这些说话人的嵌入向量可用，TS-VAD 就能同时估计所有说话人的活动状态，如图 1 所示。研究表明，这种联合估计是取得高水平说话人日志性能的关键。它不依赖注册语句，而是估计待进行说话人日志分析的录音中各说话人的档案，依据估计出的单说话人区域生成。后来研究表明，只要能够给出可能出现的说话人数上限 [20], ，就不必精确知道说话人数；而 [21]的注意力方法甚至可以免去这一要求。

然而，TS-VAD无法提供时频分辨率的活动信息，而这对于从重叠语音的混合语音中提取各个说话人的信号非常理想。我们在此提出一种说话人日志与分离的联合方法，该方法是TS-VAD架构的扩展：其最后一层被复制 F 次，其中 F 是频率箱的数量，见图1。通过这一修改，系统现在能够生成时频掩码。我们将这一方法称为目标说话人分离（TS-SEP）。

虽然这看似只是对原始TS-VAD架构的一项小改动，但其影响十分深远：它产生了一个在共同目标函数下训练的联合说话人日志与分离系统。

迄今为止，尚未实现过这种利用神经网络（NN）进行说话人日志与分离的联合处理方式，即同时考虑所有说话人。现有研究要么没有针对分离对所有说话人进行联合估计 [12], ，而这正是 TS-VAD 成功的关键；要么仅在短录音上进行实验 [22]–[24] ，这类场景不需要说话人日志，因此尚不清楚这些方法在说话人众多的长数据上效果如何。

下文首先在第二节介绍信号模型和符号。随后在第三节中，我们将介绍 TS-VAD 架构，并以此为基础在第四节推导出所提出的 TS-SEP。第五节将详细介绍实验评估。值得注意的是，我们在 LibriCSS数据集上取得了新的最先进词错误率。

II. 信号模型

我们假设观测 $\mathbf{y}_\ell$ 是 K 说话人信号$\mathbf{x}_{\ell,k}$ 与噪声信号 $\mathbf{n}_\ell$之和：

$$\mathbf{y}_\ell = \sum_k \mathbf{x}_{\ell,k} + \mathbf{n}_\ell \in \mathbb{R}^D, \tag{1}$$

其中， ℓ 是样本索引，$\mathbf{y}_\ell$、$\mathbf{x}_{\ell,k}$和$\mathbf{n}_\ell$ 均为维度为 D的向量，其中 D为麦克风总数。我们用 d表示麦克风索引，例如，$y_{\ell,d}$ 是$\mathbf{y}_\ell$的第 d个麦克风对应的元素。请注意，我们也会讨论单通道情形，此时 $D = 1$。由于我们考虑的是长时间音频录音场景，假设说话人信号$\mathbf{x}_{\ell,k}$ 具有活动与非活动时间区间是合理的。我们通过引入说话人活动变量$a_{\ell,k} \in \{0, 1\}$来明确表示这一点：

$$\mathbf{y}_\ell = \sum_k a_{\ell,k}\mathbf{x}_{\ell,k} + \mathbf{n}_\ell. \tag{2}$$

请注意，$a_{\ell,k}$ 对该方程没有影响，因为当 $a_{\ell,k}$ 为零时，$\mathbf{x}_{\ell,k}$也为零。转到短时傅里叶变换（STFT）域，我们有

$$\mathbf{Y}_{t,f} = \sum_k A_{t,k}\mathbf{X}_{t,f,k} + \mathbf{N}_{t,f}, \tag{3}$$

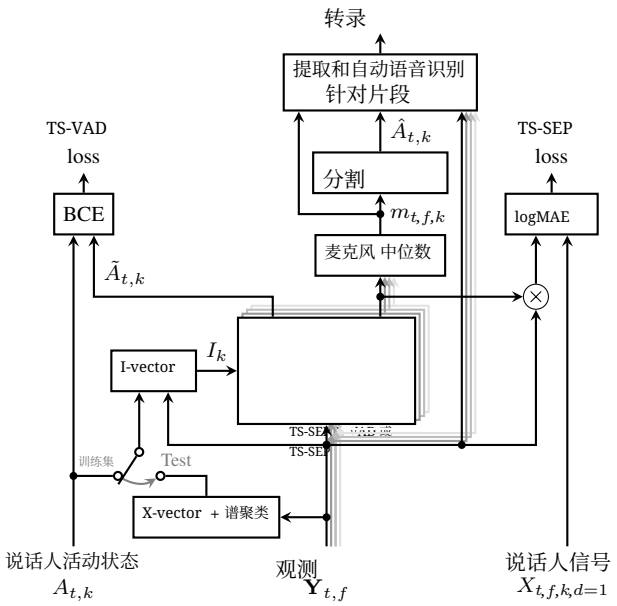


图 2. 系统概览：使用 TS-VAD损失进行预训练，使用 TS- SEP 损失进行训练，并在推理时预测转录文本。麦克风以“阴影”表示。

其中，$t \in \{1,\ldots,T\}$和 $f \in \{1,\ldots,F\}$分别是时间帧和频率箱的索引，T 和 F 分别是时间帧和频率箱的总数；$\mathbf{Y}_{t,f}$、$\mathbf{X}_{t,f,k}$和$\mathbf{N}_{t,f}$ 分别是$\mathbf{y}_\ell$、$\mathbf{x}_{\ell,k}$和$\mathbf{n}_\ell$的短时傅里叶变换。此外，$A_{t,k} \in \{0,1\}$ 表示第 k 位说话人在帧 t 的活动状态，它由 $a_{\ell,k}$ 经过时间量化至时间帧分辨率后得到。

III. TS-VAD

在 CHiME-6 挑战赛中，目标说话人语音活动检测（ TS-VAD） [8], [18]在说话人日志任务中取得了令人瞩目的结果，大幅超越了其他方法。其基本思想与个人语音活动检测 [19]类似：训练一个神经网络（NN），根据表示说话人的嵌入 I_k ，从混合语音 $\mathbf{Y}_{t,\ f}$中预测目标说话人的语音活动 $A_{t,\ k}\,k$ ，如图 1 所示。

TS-VAD 与先前的工作有两个主要区别：首先，如图 2 所示，说话人嵌入是从录音中估计得到的，而不是使用注册语句；其次，如图 1 所示，组合层在给定每位说话人的嵌入后，同时估计一个片段中所有说话人的活动状态。

A. 说话人嵌入估计

TS-VAD 依赖初始说话人日志信息，例如人工标注，或由嵌入提取系统（如 X-vector [25]）提供的信息，随后采用聚类方法（如谱聚类 [26]）。然后，从仅有单一说话人活动的片段中计算说话人嵌入向量（如 I 向量 [27]）

$$I_k = \mathrm{Emb}\Big\{y_{\ell,d=1}, \forall\ \ell \text{ s.t. } \hat{a}_{\ell,k}=1 \ \&\ \sum_{\tilde{k}\neq k}\hat{a}_{\ell,\tilde{k}}=0\Big\} \in \mathbb{R}^E. \tag{4}$$

这里，Emb$\{\cdot\}$ 表示从第一个麦克风采集的语音片段中计算说话人嵌入向量，其中仅有单一说话人活动。此外，$\hat{}$ $a_{\ell,\ k}$ 是$a_{\ell,\ k}$的估计值。

虽然该方法不需要注册语音，但它使 TS-VAD 依赖于另一个初始的说话人日志系统。因此，TS-VAD 可以被视为一种细化系统：给定第一次说话人日志的估计值，应用 TS-VAD 来估计更好的说话人日志。在 [18],中报告称，对于 TS-VAD，I-vector 嵌入优于 X-vector，这与其他出版物的结论相反；后者认为，在说话人分离任务中，X-vector 优于 I-vector[25]。我们遵循 [18]并使用 I-vector。

*B. TS-VAD*架构

TS-VAD网络由三个组件组成，组件之间进行堆叠操作，如图 1 所示。首先，将混合语音中一个麦克风的对数谱图特征和对数梅尔滤波器组特征进行堆叠，并通过若干说话人无关层进行编码。这些层可以视为特征提取层，将其编码为维度为 $\mathbb{R}^{T\times Z_1}$的矩阵，其中 Z_1 是单个逐帧表示的大小。

接下来，对于每位说话人 k，将逐帧表示与该说话人的说话人嵌入拼接起来，得到第二网络组件的输入，其维度为 $\mathbb{R}^{T\times(Z_1+E)}$。第二网络组件分别处理每位说话人，但说话人之间共享神经网络层参数。因此，只能通过说话人嵌入而非神经网络层参数来区分说话人。对于每位说话人，第二网络组件的输出维度为 $\mathbb{R}^{T\times Z_2}$，其中 Z_2 表示每帧的大小。到此为止，神经网络的设计与个人语音活动检测 [19] 相同。

在最后几层神经网络之前，将所有说话人的隐藏特征拼接起来，得到一个维度为 $\mathbb{R}^{T\times(K\cdot Z_2)}$ 的大型特征矩阵。最终几层生成维度为 $\mathbb{R}^{T\times K}$ 的输出，经过阈值处理和平滑处理（见ˆ 第 IV-C 节）后，得到所有说话人k 在所有帧 t 上的活动估计 $A_{t,k}$ 。这种联合估计使得区分相似说话人更加容易。事实上， [18] 的研究表明，末尾的联合估计层对于获得良好性能至关重要。

从这一描述可以明显看出，TS-VAD假定已知待进行说话人分离的会议中说话人的总数 K ，因为 K 决定了网络输出的维度。通过引入注意力机制，可以放宽这一限制；在完全重叠语音分离的情况下， [21], 已展示了这一方法。不过，我们保留原始的TS-VAD堆叠结构，仅将说话人数量从 [18]中的4人增加到8人。这样做首先是因为，对于许多会议而言，将8名说话人作为上限已经足够大；其次是为了便于与 [20], 进行更好的比较，因为其中也采用了这种堆叠结构。

C. 从单通道到多通道

当有多通道可用时，可以采用不同的方法来利用这些通道，例如通过通道间相位差（IPD）等空间特征 [28]，进行中间信号降噪，如注意力层中的方法 [18]，，以及输出信号降噪 [29]。这些方法各有不同的优点和缺点。本文采用输出信号降噪：这意味着对每个通道分别应用神经网络，并通过中值运算将输出约简为单一估计值。以这种方式引入多通道后，我们可以使用单通道训练神经网络，并在测试集阶段使用所有通道。除此之外，这使我们的方法不受阵列几何结构和麦克风数量的影响；在CHiME-6 [1]，中，移动说话人的存在也变得不那么关键。最后，已有研究表明，该方法对于模拟训练数据与真实测试数据之间的不匹配具有鲁棒性 [29]。

为简化符号表示，在以下部分中，我们省略神经网络输入和输出的麦克风索引。训练时，我们仅使用一个参考麦克风；测试时，神经网络分别独立应用于每个麦克风，然后取中位数。

IV. 从 TS-VAD 到 TS-SEP

下面将介绍为实现联合说话人日志与分离而对TS-VAD架构所做的修改。说话人日志与（基于掩码的）分离之间的关键区别在于，说话人日志的输出具有时间分辨率，而分离则需要时频分辨率。

A. 从帧到时频分辨率

从 TS-VAD 系统出发，将最后一层的输出大小从 K 个说话人改为 K 个说话人乘以 F 个频率箱：$\mathbb{R}^{T\times K} \to \mathbb{R}^{T\times(F\cdot K)}$，其中 $\to$表示“重复”操作，即重复最后一层的权重。将输出从 $\mathbb{R}^{T\times(F\cdot K)}$重新排列为$\mathbb{R}^{T\times F\times K}$后，我们便为每个说话人 k得到一个 $(T \times F)$ 维的时频输出。经过 Sigmoid 非线性函数处理后，可确保数值处于 $[0, 1]$，，因此该输出可解释为掩码 $m_{t,\ f,\ k},\ \forall t,\ f,\ k$，可用于声源提取，例如通过掩蔽实现：

$$\hat{X}_{t,f,k} = m_{t,f,k} Y_{t,f,d=1}, \tag{5}$$

其中 $d = 1$ 是参考麦克风。我们将该系统称为目标说话人分离（TS-SEP）。

图 1 顶部的示意图展示了 TS-VAD 和 TS-SEP 的不同输出，图 2 概述了 TS-SEP，其组件将在下文介绍。

B. 训练计划与目标

由于 TS-SEP 源自 TS-VAD，一个显而易见的训练选择是采用两阶段训练计划：在第一阶段，即预训练阶段，先训练一个常规的 TS-VAD 系统，直到其开始区分不同说话人；然后，将最后一层复制 F 次，以获得所需的时频分辨率；随后第二个训练阶段则是对

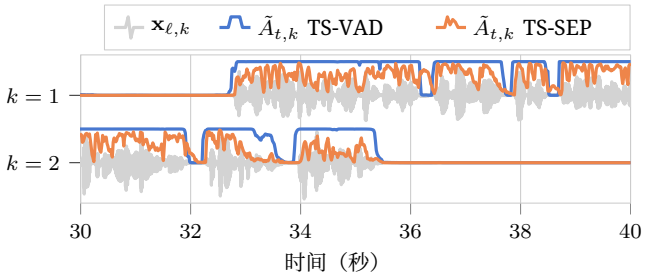


图 3. 来自 LibriCSS DEV OV40 数据集的 8 位说话人中 2 位的估计说话人活动状态 $\tilde{A}_{t,\ k}$，以及相应的干净说话人时域信号 $\mathbf{x}_{\ell,\ k}$。建议以彩色模式查看。

TS-SEP 系统进行训练，并使用 TS-VAD 系统的参数进行初始化。

对于 TS-VAD预训练，我们采用了原始论文提出的训练损失 [18]，，即所有说话人的估计活动状态与真实活动标签之间二元交叉熵（BCE）损失之和。

TS-SEP 系统的训练目标有多种选择。我们选择了时域重建损失，因为它隐式地考虑了相位信息（关于时域与频域重建损失的讨论，参见[30]，[31] ）。具体而言，时域信号重建损失通过对$X_{t,f,k}$ 应用逆短时傅里叶变换ˆ，得到ˆ $x_{\ell,k}$ ，再将其与真实值$x_{\ell,k}$进行比较，计算对数平均绝对误差（LogMAE）：

$$\mathcal{L} = \log_{10} \frac{1}{L} \sum_{k} \sum_{\ell} \left| \hat{x}_{\ell,k} - x_{\ell,k} \right|. \tag{6}$$

显然，也可以采用其他损失函数，例如 MAE、均方误差（MSE）或信号失真比（SDR）。不过，需要注意的是，对于包含对数运算的重建损失，应在执行对数运算之前先对各说话人的损失求和。否则，如果某个说话人完全静音，损失将无法定义，训练也可能变得不稳定 [32]。

我们还尝试从头开始训练 TS-SEP 系统。然而，事实证明，这种做法对训练损失的选择敏感得多。我们发现，只有改进的二元交叉熵损失能够从头开始学习：该损失人为地提高了与“目标存在”对应的损失项的权重；其他选择则完全无法收敛[1]。但即使采用这种损失，其最终性能也不如两阶段训练[2]。因此，这里我们只报告采用上述两阶段训练的实验。

C. 活动估计 / 分割

TS-SEP 输出具有时频分辨率的掩码。为了得到类似 TS-VAD、只有时间分辨率的估计值，我们取

[1]网络最初学会的只是预测全零，因此无法摆脱这些局部最优解。

[2]两阶段训练的排接最小排列词错误率（cpWER）为 72%，而使用 LogMAE 损失从头开始训练时，cpWER 为 100%（原因是零估计值），使用基于二元交叉熵的损失时，cpWER 为 12.0%。

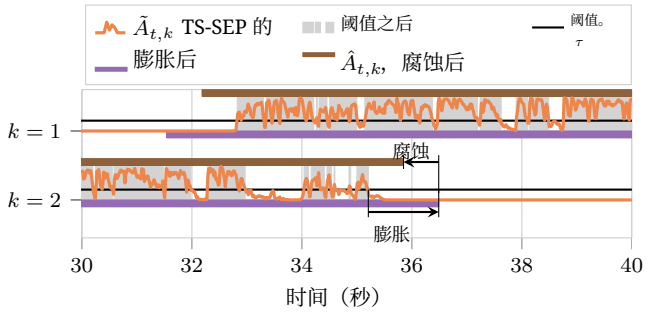


图 4。对$A_{t,\ k}$ 进行阈值处理和“闭运算”形态学操作（膨胀后˜ ˆ腐蚀），以获得$A_{t,\ k}$；示例为 LibriCSS DEV OV40 数据集中的 8 位说话人中的 2 位。建议彩色查看。

跨频率的平均值：

$$\tilde{A}_{t,k} = \frac{1}{F}\sum_{f} m_{t,f,k}. \tag{7}$$

通常，该估计值在每个词处都有尖峰，而在词与词之间接近于零，见图 3。为了填补这些间隙，我们遵循[6] ，使用了阈值处理和“闭运算”形态学操作（见图 4），这在图像处理中是已知的方法 [33]：首先进行膨胀操作，将滑动窗口以一个采样点的步长移过信号，并将窗口内的最大值作为中心采样点的新值；随后进行腐蚀操作，但改用最小值操作，从而得到最终平滑的活动估计值

$$\hat{A}_{t,k} = \mathrm{Erosion}(\mathrm{Dilation}(\delta(\tilde{A}_{t,k} \geq \tau))). \tag{8}$$

这里，τ 是阈值，δ（x）表示克罗内克δ函数：当 x 为真时取 1，否则取 0。当膨胀窗口大于腐蚀窗口时，语音活动会被高估，即将两者之差除以二后分别加到片段的起始端和末端，如图 4 所示。虽然高估可能会降低说话人日志性能，但对于自动语音识别（ASR）而言，情况未必如此：实际上，使用包含单个话语以及录音开头和结尾处少量静音（或背景噪声）的录音来训练自动语音识别系统是很常见的。我们将在第五节-H中进一步研究这种高估。

ˆ 在 $A_{t,k}$ 的活动状态变化点处，说话人信号会被切分，从而形成说话人活动状态恒定的片段。

这种阈值处理和“闭运算”流程不同于 TS-VAD 通常采用的基于中位数的平滑处理。当说话人分离系统经过训练，能够填补短暂非活动状态的间隙（例如词语之间的停顿）时，基于中位数的平滑处理是合适的。但在 TS-SEP 中，系统经过训练，能够为每个说话人预测时频重建掩码，因此掩码值通常小于 VAD 值；短暂的非活动状态会产生零值，而重叠区域的值则小于非重叠区域。阈值处理和闭运算等形态学操作有助于填补这些间隙。

D. 提取

在评估时，通过将掩码与输入语音的短时傅里叶变换相乘，可获得说话人 k 的信号估计值，见公式（5）。如果有多通道输入，则源提取时可采用掩码乘法的替代方法：利用估计的掩码计算波束形成器系数 [29]。对于波束形成，目标信号的空间协方差矩阵

$$\mathbf{\Phi}_{xx,f,b} = \frac{1}{|\mathcal{T}_b|}\sum_{t\in\mathcal{T}_b} m_{t,f,k_b}\mathbf{Y}_{t,f}\mathbf{Y}_{t,f}^{\mathrm{H}} \tag{9}$$

以及失真

$$\mathbf{\Phi}_{dd,f,b} = \frac{1}{|\mathcal{T}_b|}\sum_{t\in\mathcal{T}_b} \max\left(\varepsilon, \sum_{\tilde{k}\neq k_b} m_{t,f,\tilde{k}}\right)\mathbf{Y}_{t,f}\mathbf{Y}_{t,f}^{\mathrm{H}} \tag{10}$$

首先计算。这里，b 表示片段索引，$\mathcal{T}_b$ 是属于片段 b 的帧索引集合，k_b 是在片段 b 中处于活动状态且需要提取的说话人的索引 [3]，ε =0.0001 是为保证稳定性而引入的一个小值，而 $\sum_{\tilde{k}\neq k_b}$ 表示对除 k_b 之外的所有说话人索引求和。

利用这些协方差矩阵，可以计算波束形成器系数，例如使用 [34] 所述形式的最小方差无失真响应（MVDR）波束形成器：

$$\mathbf{w}_{f,b} = \mathrm{MVDR}(\mathbf{\Phi}_{xx,f,b}, \mathbf{\Phi}_{dd,f,b}). \tag{11}$$

最后，将波束形成器应用于输入语音以提取声源：

$$\hat{X}_{t,f,k_b} = \mathbf{w}_{f,b}^{\mathrm{H}}\mathbf{Y}_{t,f}, \quad \forall\, t \in \mathcal{T}_b. \tag{12}$$

在我们的实验中，我们还将波束形成与掩码乘法结合起来 [35]，从而更好地抑制了竞争说话人：

$$\hat{X}_{t,f,k_b} = \mathbf{w}_{f,b}^{\mathrm{H}}\mathbf{Y}_{t,f}\max(m_{t,f,k_b}, \xi), \quad \forall\, t \in \mathcal{T}_b, \tag{13}$$

其中，$\xi \in [0,1]$ 是掩码的下限/阈值。

下限的依据是 [36]中的观察结果：与不做任何处理相比，纯神经网络语音增强可能会降低自动语音识别性能；将观测的缩放版本添加到增强后的信号中，则可能使自动语音识别性能优于不做任何处理。

E. 引导式声源分离（*GSS*）

直接使用掩码进行提取的另一种做法是先对掩码进行微调。如果有多通道输入，可以通过利用空间信息的 SMM 来实现。此处，我们采用 [7], 中提出的 GSS，其中引导信息由说话人日志信息 $A_{t,\ k}$ 提供：只有当 $A_{t,\ k}$ 为 1 时，第 k 个混合语音组件在帧 t 中处于活动状态的后验概率才可能非零。SMM 应用于片段 b，这些片段的边界

[3]片段取自 ˆ$A_{t,k}$，使得每个片段 b都与一个目标说话人 k_b相关联；当多个说话人同时处于活动状态时，每个说话人都会获得不同的片段索引 b。

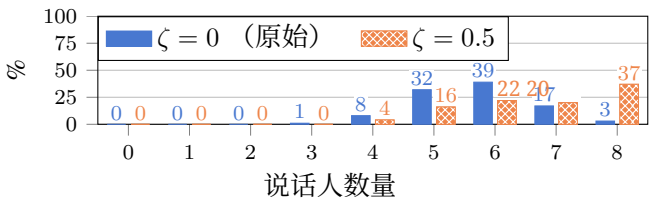


图 5. 训练数据中 1 分钟片段内活动说话人数的分布，取决于使用叠加/Mixup 的概率 ζ （$\zeta = 0$ 对应于未使用叠加的原始数据集）。

由 $A_{t,\ k}$ 以及一些上下文确定。利用引导信息进行的 EM 迭代会辅以一个无引导信息的 EM 步骤。最终估计的类别后验概率构成说话人专属时频掩码，用于通过波束形成提取声源。有关 GSS 的详情，请参阅 [7]。

以往的工作会通过沿频率轴广播语音活动检测（VAD）信息来初始化 SMM，该信息可来自人工标注（CHiME-5 [37]），也可由自动方式确定（CHiME-6 Track2 [1]）；而我们在此提出使用 $m_{t,f,k}$ 来初始化 SMM 的后验，这一信息仅 TS-SEP 可用；语音活动信息 $A_{t,k}$ 则 TS-VAD 和 TS-SEP 均可用。

第五节 实验

A. 训练集和测试数据集

说话人日志和识别实验在 LibriCSS数据集上进行，该数据集广泛用于评估会议识别系统 [3]。它包含对 LibriSpeech 句子进行扬声器播放并重新录制的语音，这些语音以反映典型会议场景的方式进行混合。该数据集由 10 个时长为一小时的会话组成，每个会话又细分为 6 个时长为 10 分钟的迷你会话，其重叠比例各不相同，范围为 0% 至 40%。每个会话中共有 8 名说话人处于活跃状态。录音设备为七通道圆形麦克风阵列。按照 [2], ，我们将第一个会话用作开发集（DEV），其余 9 个会话用作评估集（EVAL）。

作为训练数据，我们采用了 LibriCSS 作者通过脚本[4] 生成的模拟会议数据。虽然每个训练会议包含 8 名说话人，但我们仅使用时长为 1 分钟的片段进行训练。由于在一个片段内通常只有较少的说话人处于活跃状态，如图 5 所示，我们将语音片段进行叠加，以人为增加活跃说话人的数量，相关讨论见第五节-D。

B. 性能指标

由于本文聚焦于融入说话人何时发言信息的会议转录，因此我们同时评估说话人日志和词错误率的表现。

作为说话人日志性能指标，我们采用广泛使用的说话人分离错误率（DER），该指标通过美国国家标准与技术研究院的“md-eval-22.pl”脚本测量 [38]。DER 是虚警、漏检和说话人混淆错误的总和除以说话人总时长。

对于多个说话人的部分重叠语音识别，有多种词错误率（WER）定义 [39]。由于 TS-VAD 会将同一说话人的话语归为一组，我们主要报告拼接最小排列词错误率（cpWER）：将每位说话人的所有转录文本拼接起来，并针对估计转录文本与不同说话人的真实词序列之间的映射计算词错误率，取其中词错误率最低的结果。

请注意，cpWER 也会统计由说话人日志错误导致的识别错误。例如，如果说话人日志将某个话语的说话人身份标错，那么该错误会计入正确说话人转录文本的删除错误数；此外，无论识别出的词语是否正确，该错误还会计入被错误识别为说话人的转录文本的插入错误数。

由于我们关注说话人分离错误所造成的影响，因此还计算了将识别出的片段转录文本指派 k_b 到真实转录流的方式，使 cpWER 尽可能小。换言之，对于每个片段 b，我们使用使 cpWER 最小的说话人标签，而不是从$A_{t,\ k}$中得到的估计^说话人标签 k_b ，同时保留估计的片段起止时间。这与 [39]，中定义的 MIMO-WER 类似，但真实值与估计值互换。以这种方式计算的错误率忽略了说话人分离错误的影响，因此我们称之为说话人分离不变的串联最小置换词错误率（DI-cpWER）。因此，cpWER 与 DI-cpWER 之差可解释为说话人分离错误造成的错误。

在相关文献中，常使用 asclite 工具计算多说话人转录文本 [40]的词错误率。它同样忽略说话人分离错误，但仅适用于较短的片段，且估计的“说话人”数量较少。过去曾将其用于 LibriCSS，不过当时使用的是所谓的连续语音分离（CSS）系统，该系统将数据分离为两个流，每个流都包含若干位说话人的信号，且这些说话人之间没有重叠 [3]。在我们的设置中，我们生成八个流，会议中的每位说话人对应一个流。由于 asclite 的计算和内存需求会随流的数量呈指数级增长，因此在我们的系统中使用 asclite 在计算上不可行。

asclite 和 DI-cpWER 都会计算一种忽略说话人分离错误的词错误率（WER）。因此，所得数值大致可以相互比较，我们将说话人无关词错误率（WER）作为这类忽略说话人分离错误的词错误率的统称。

我们使用 ESPnet 框架中的两种不同的预训练自动语音识别系统 [41]来估计转录结果。两者均直接使用，未经任何微调。第一种系统简称为“基础版”[5] [42], ，采用 Transformer 架构。该系统针对 LibriSpeech 设计并训练，在 LibriSpeech 干净测试集上的词错误率为 2.7%。第二种系统则

[4]https://github.com/jsalt2020-asrdiar/jsalt2020 模拟

[5]https://zenodo.org/record/3966501

简称为“WavLM”[6] [43]，，是一种基于 Conformer、利用 WavLM [44]特征的自动语音识别系统。该系统针对 CHiME-4 [45] 设计，并使用 Libri-Light [46]，、GigaSpeech [47]，、VoxPopuli [48]，、CHiME-4 [45]，和 WSJ0/1 [49] 进行训练。在 LibriSpeech 干净测试集上，其词错误率为 1.9%。

除非另有说明，以下表格中的结果均使用 LibriCSS 数据集的全部七个通道获得。请注意，每个通道的掩码都是独立估计的，然后对各通道取中位数，以获得最终掩码，见图 2。

C. 预处理

语音数据经 64 ms 窗长和 16 ms 帧移的短时傅里叶变换转换到短时傅里叶变换域。在短时傅里叶变换域中，将对数频谱图和梅尔频率倒谱系数（MFCC）堆叠，作为神经网络的输入特征。

计算说话人嵌入时，我们遵循了 [18]：训练时，使用预言机标注识别单说话人区域，并据此估计 I 向量；测试时，我们使用 X 向量提取器和 ESPnet LibriCSS 配方 [41] 中的谱聚类（SC）来估计标注，再用这些估计结果提取 I 向量[7]。请注意，这一初始估计无法识别重叠语音，因此计算嵌入时无法排除重叠语音。

在 Kaldi [50], 中，因此在 TS-VAD [18], 的原始实现中也一样，通常会为每个数据集计算一个均值和方差向量，并用它们对输入特征向量进行归一化。我们最初跳过了这一步归一化，但发现若不进行这种领域自适应，性能会严重下降，词错误率（WER）从 7.7% 增至 21.5%。因此，在此报告的所有实验中，我们都针对输入特征和 I- 向量，在模拟验证数据集与重新录制的 DEV 和 EVAL LibriCSS 数据集之间进行了矩匹配（均值和方差匹配）。

作为额外的增强，在测试时，神经网络（NN）和提取之前进行了基于加权预测误差（WPE）的去混响 [51], [52] 。

D. 训练细节

在开发该系统的过程中，我们发现有几个方面对收敛特性或最终性能的影响超出预期，例如将网络开始学习所需的时间从数周缩短到数小时。我们不会为每项有用的改进提供详细的实验结果，但希望分享这些发现，以供社区参考。

我们使用一分钟长的数据块进行训练。为了降低对人为说话人排序的依赖，我们按照 [18] 的方法，在输入端随机排列说话人嵌入，并在神经网络之后还原排列。

[6]https://huggingface.co/espnet/simpleoierlibrispeech_自动语音识别_训练集 自动语音识别_Conformer7 WavLM large raw en 字节对编码5000 sp [7]https://kaldi-asr.org/models/13/0013 librispeech v1 extractor.tar.gz

表 I 使用不同的说话人分离信息对观测进行分段，并应用不同的自动语音识别系统（不对重叠语音进行中间分离），比较我们的 TS-VAD 实现与 [20], 的实现。使用 LibriCSS 的第一个麦克风。

	说话人识别参考说话人分离	ASR	DER 开发集	DER 评估集	cpWER 开发集	cpWER 评估集
(1)	[20]理想	隐马尔可夫模型-深度神经网络	–	–	–	23.1
(2)	[20]TS-VAD	隐马尔可夫模型-深度神经网络	–	7.6	–	25.8
(3)	[44]理想	WavLM	–	–	–	6.0
(4)	理想	Base	0.0	0.0	19.17	19.77
(5)	SC	Base	18.93	17.93	29.07	27.77
(6)	TS-VAD	Base	6.30	5.65	21.55	21.27
(7)	TS-VAD	WavLM	6.30	5.65	10.39	9.13
(8)	理想	WavLM	0.0	0.0	6.79	6.01

此外，我们发现，在最后一个非线性函数之前，使用不同的排列运行神经网络两次，并在还原排列后对输出取平均，可以使训练收敛得更好。 [18] 中提到，在一些初步实验的评估中使用了这一方法。

对于分离系统，我们发现，在训练初期存在更多重叠时，收敛特性更好。对于基于 TS-VAD 的说话人日志，我们也观察到了这一现象，因此采用了 [53], 中的叠加思路，该方法与 Mixup [54] 类似，并以 $\zeta = 50\%$ 的概率，将一分钟的数据与紧接着的下一分钟数据相加作为训练数据，而不是使用原始数据。为了计算这种叠加对应的 TS-VAD 目标，我们使用了逻辑或运算符。尽管这种叠加造成了训练集与测试集之间的不匹配（例如，自重叠以及同时活跃和总活跃说话人数量增加），但我们观察到收敛效果更好，最终性能也有所提升。图 5 展示了采用和未采用叠加时训练数据中活跃说话人数量的分布。

E. 后处理 / 分割

对于名为“base”的预训练自动语音识别系统，我们发现长片段[8], 会出现异常行为，因此我们按照 $A_{t,k}$，在静音位置拆分较长的˜活动，使每个片段都不超过 12 秒。最小分段长度设为 40 帧（0.64 秒），但只有在未使用高估时才有必要。

F. 基线系统与参考结果

TS-VAD [18] 针对 CHiME-6 数据的公开源代码使用 Kaldi 编写，而针对 LibriCSS[20] 的修改版本并未公开。我们使用 PyTorch [55] 重新实现了 TS-VAD，并尽可能忠实地遵循原始实现。例如，对于神经网络层，我们在三个神经网络模块中分别使用一个双向长短期记忆投影网络[9] [57],两个双向长短期记忆投影网络，以及一个双向长短期记忆投影网络后接一个前馈层。

[8]例如，转录内容突然被一串由字母 E、O 和 T 组成的单字母词打断。9

双向长短期记忆投影网络是一种双向长短期带有前馈投影层的长期记忆 [56] 循环神经网络。

表 II 使用分割超参数 THR.$\tau = 0.6$、说话人漏检数 $= 161$ 和说话人误报数 $= 81$ 时，TS-SEP 采用各种增强策略的结果。

ID	WPE	提取 BF	掩蔽（ξ）	DER 开发	DER 评估	cpWER 开发	cpWER 评估
(1)	–	–	–	15.47	13.89	26.53	24.42
(2)	✓	–	–	15.35	13.68	26.26	24.08
(3)	✓	–	√(0.0)	15.35	13.68	12.42	10.49
(4)	✓	–	√(0.5)	15.35	13.68	22.99	21.07
(5)	✓	✓	–	15.35	13.68	10.51	8.82
(6)	✓	✓	√(0.0)	15.35	13.68	11.08	9.29
(7)	✓	✓	√(0.01)	15.35	13.68	11.12	9.26
(8)	✓	✓	√(0.05)	15.35	13.68	10.93	9.00
(9)	✓	✓	√(0.2)	15.35	13.68	10.43	8.48
(10)	✓	✓	√(0.5)	15.35	13.68	**10.16**	**8.42**
(11)	–	✓	√(0.5)	15.47	13.89	11.31	9.46

表 III 使用 WPE+BF+掩蔽（$\xi = 0.5$）作为增强时，TS-SEP 在不同分割超参数下的结果。

ID	分割 阈值 τ	说话人漏检数	说话人误报数	DER 开发集	DER 评估集	cpWER 开发	cpWER 评估
(1)	0.3	201	81	24.91	21.94	10.41	8.26
(2)	0.3	161	81	17.23	14.69	10.06	7.72
(3)	0.3	121	81	9.89	7.89	9.99	7.70
(4)	0.3	81	81	7.61	6.49	12.98	11.16
(5)	0.1	161	81	19.55	16.46	10.36	7.98
(6)	0.2	161	81	18.22	15.38	10.27	7.79
(8)	0.3	161	81	17.23	14.69	10.06	7.72
(9)	0.4	161	81	16.44	14.16	**9.95**	**7.72**
(10)	0.6	161	81	**15.35**	**13.68**	10.16	8.42
(11)	0.3	141	61	**17.14**	**14.61**	**10.01**	**7.72**
(12)	0.3	161	81	17.23	14.69	10.06	7.72
(13)	0.3	181	101	17.41	14.85	10.06	7.75

图 1. 尽管如此，它们在某些细节上可能有所不同。因此，我们首先验证我们的实现是否具有竞争力。从表 I 可以看出，我们实现的 DER 和词错误率与原始实现相近（第 (6) 行与第 (2) 行）。TS-VAD 的效果优于谱聚类（第 (6) 行与第 (5) 行）；结合基于 WavLM 的自动语音识别系统，我们在未进行任何增强的情况下取得了最佳词错误率（第 (7) 行）。

G. 提取与增强

在表 II 中，我们分析了所提出的 TS-SEP 在各种提取策略下的性能。TS-SEP 会计算一个掩码，从观测中提取说话人的源信号。我们首先不进行增强（第 (1) 行），以考察说话人日志性能。请注意，在表 II 中，我们使用了 161 帧的膨胀，而表 I 中使用的是 81 帧。由此产生的高估导致 DER 较高，且与 TS-VAD 相比，cpWER 略有恶化（原因将在下一节解释）。加入基于 WPE 的去混响后，性能略有提升（第 (2) 行）；使用掩蔽进行增强则使 cpWER 大幅降至 10.49%（第 (3) 行）。采用波束形成（第 (5) 行）后，我们进一步将 cpWER 降至 8.82%。在波束形成后进行掩蔽（第 (6) 行）则导致结果变差，这可能是因为掩蔽引入了伪影。不过，通过为掩码选择合适的下阈值 $\xi = 0.5$，我们实现了小幅提升，cpWER 达到 8.42%（第 (10) 行）。我们还列出了第 (11) 行，以帮助说明 WPE 的影响。

下阈值的效果与 [36]语音增强[10]中的观察结果一致：对于自动语音识别，掩蔽产生的伪影会带来负面影响，但为掩码精心选择较低的下阈值可以提升自动语音识别性能。

[10]在 [36], 中，将观测的缩放版本加到估计值上；在我们的系统中，这相当于给掩码加上一个常数。请注意，我们不会加上观测，因为其中包含跨说话人干扰，而波束形成信号已被证明适用于自动语音识别 [29]。

H. 分段

在此未报告的初步实验中，我们优化了分段参数，以便在使用 TS-VAD 系统时将 DER 降至最低。使用 TS-SEP 系统时，情况有两点变化。首先，高估活动区间长度对声源提取而言可能并非关键问题，因为增强可以抑制边界处可能处于活动状态的串扰说话人。此外，高估会产生与自动语音识别系统典型训练数据更匹配的话语边界，如 IV-C 节所述。其次，TS-SEP 经过训练，旨在预测重建掩码，因此活动状态$\hat{}$估计值 $A_{t,\ k}$ 通常小于 TS-VAD 的估计值，尤其是在说话人不活动或活动程度较低的时频单元中，如图 3 所示。另一方面，TS-VAD 系统经过训练，旨在使话语的整个持续时间内的值都接近 1。鉴于这种训练失配，我们预计 TS-SEP 会倾向于在公式 (8) 中采用较小的阈值 τ。

我们在表 III 的实验中验证了这两种假设。优化分割超参数时，DER 和 WER 表现出相反的趋势。对于 DER，正如预期，高估会产生负面影响（第 (4) 行与第 (1) 至 (3) 行相比）。相反，对于 WER，正如上一段所述，适度高估会有所帮助。初步实验中，$\tau = 0.6$ 的阈值对于 TS-VAD 是一个不错的选择；而对于 TS-SEP，较小的取值（范围为 0.3 至 0.4）带来了最佳表现（第 (5) 至 (10) 行）。同时扩大或缩小膨胀和腐蚀窗口，影响都很小（第 (11) 至 (13) 行）。

I. 单通道系统

在表 IV 中，我们将系统中的组件替换为对应的单通道版本，从而构建出单通道系统：波束形成、掩蔽、去混响和领域自适应（DA）。波束形成显然需要多个通道，因此单通道替代方案就是不使用任何波束形成。相比之下，声源提取的掩码既可以根据单个（参考）通道计算，也可以采用多通道方式计算：对每个通道分别应用掩码估计神经网络，再将得到的

表 IV 单通道系统。√ 表示组件为单通道，– 表示组件使用多通道。DA 表示领域自适应，其中区分了自适应是对所有通道同时进行（–），还是按通道分别进行（√）。

ID	WPE	Mask 降噪	提取	cpWER DEV 领域自适应：–	cpWER DEV 领域自适应：√	cpWER EVAL 领域自适应：–	cpWER EVAL 领域自适应：√	DI-cpWER DEV 领域自适应：–	DI-cpWER DEV 领域自适应：√	DI-cpWER EVAL 领域自适应：–	DI-cpWER EVAL 领域自适应：√
(1)	–	–（中位数）	–（BF+掩蔽（$\xi = 0.5$））	10.06	9.83	7.72	7.70	5.41	5.12	5.92	5.85
(2)	–	√（参考）	–（BF+掩蔽（$\xi = 0.5$））	8.96	8.45	9.05	7.80	5.94	5.37	6.59	5.98
(3)	–	–（中位数）	√（掩蔽（$\xi = 0.0$））	12.14	11.90	10.11	10.00	7.55	7.27	8.34	8.20
(4)	–	√（参考）	√（掩蔽（$\xi = 0.0$））	11.83	11.07	12.04	10.30	8.80	8.07	9.70	8.51
(5)	✓	√（参考）	√（掩蔽（$\xi = 0.0$））	13.40	12.12	13.27	11.55	10.45	9.01	10.83	9.74

表 V 不同提取方法和自动语音识别系统下 TS-VAD 与 TS-SEP 的对比，其中说话人漏检数为 $= 161$，说话人误报数为 $= 81$，阈值在 TS-VAD 中设为 $\tau = 0.6$，在 TS-SEP 中设为 $\tau = 0.3$。

ID	说话人分离/分割	提取	ASR	DER 开发	DER 评估	cpWER 开发	cpWER 评估	DI-cpWER 开发	DI-cpWER 评估
(1)	TS-VAD	BF+掩蔽 ($\xi = 0.5$)	Base	18.72	16.28	17.26	15.98	13.22	13.87
(2)	TS-VAD	GSS T 初始化	Base	18.72	16.28	9.09	6.43	4.74	4.32
(3)	TS-SEP	BF+掩蔽（$\xi = 0.5$）	Base	17.23	14.69	10.06	7.72	5.41	5.92
(4)	TS-SEP	GSS T 初始化	Base	17.23	14.69	8.93	6.32	4.23	4.54
(5)	TS-SEP	GSS TF 初始化	Base	17.23	14.69	8.80	6.15	4.07	4.37
(6)	TS-VAD	BF+掩蔽 ($\xi = 0.5$)	WavLM	18.72	16.28	12.13	9.50	7.95	7.35
(7)	TS-VAD	GSS T 初始化	WavLM	18.72	16.28	8.44	5.47	4.20	3.32
(8)	TS-SEP	BF+掩蔽（$\xi = 0.5$）	WavLM	17.23	14.69	8.55	5.76	3.85	3.92
(9)	TS-SEP	GSS T 初始化	WavLM	17.23	14.69	8.12	5.10	3.40	3.29
(10)	TS-SEP	GSS TF 初始化	WavLM	17.23	14.69	**8.11**	**5.06**	**3.32**	**3.26**

掩码通过中值运算合并为一个掩码。同样，基于 WPE 的去混响通常会对所有可用通道联合处理，但也可以对每个通道分别处理。对于领域自适应，默认情况下，我们使用模拟验证集的一个通道计算训练时统计量，并使用重新录制的测试集的所有通道计算测试时统计量。为了使其适用于单通道，我们对每个通道分别计算测试时统计量。对于开发集，我们在 cpWER[11]，中观察到一些离群值，因此这里也报告了更稳健的 DI-cpWER。

将各组件依次切换为单通道版本后，cpWER 均有所上升，只有领域自适应产生了积极效果。我们通过更多实验探究了这一令人意外的结果。结果发现，在 LibriCSS 数据中，录音由圆形麦克风阵列采集，阵列中心也有一个麦克风；作为参考麦克风的中心麦克风，其统计特性与其他麦克风明显不同。这与模拟数据不同：模拟数据采用全向麦克风特性，各麦克风之间的统计特性没有明显差异。最终，采用完全单通道系统后，我们得到 11.55% 的 cpWER 和 9.74% 的 DI-cpWER。

[11]对于说话人分离无关指标，我们选择了 DEV 集，因为 6 个录音就足够了。对于说话人分离感知指标，由于该集合规模较小，一次错误的指派可能会产生较大影响。

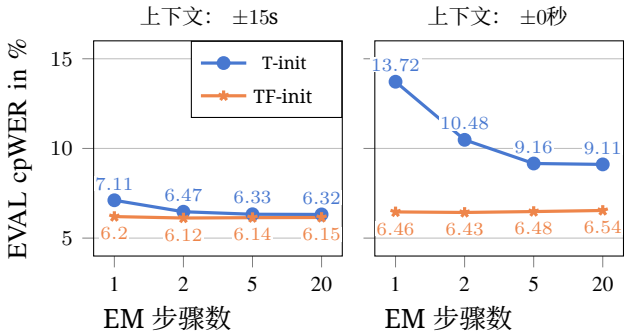


图6. 对于不同数量的引导式EM步骤和不同长度的时间上下文，使用 GSS 的 TS-SEP，其中 GSS 使用语音活动检测（T-init）或掩码初始化（TF-init）。

J. TS-VAD 与 TS-SEP 的比较及使用 GSS 进行掩码精炼

GSS 可以被视为 TS-SEP 的一种替代方案，用于将语音活动状态信息从时间分辨率扩展到时频分辨率。不过，它也可以作为一种掩码精炼操作，应用于 TS-SEP 的输出。在表 V 中，我们探讨了这两种不同的设置。

首先，我们比较了 TS-VAD 与 TS-SEP 的性能，且未使用 GSS 对二者的输出进行后处理（第（1）行和第（3）行）。可以明显看出，在 cpWER 方面，使用具有时频分辨率的掩码具有优势（7.72% 对 15.98%）。接下来，仅使用 TS-SEP 提供的时间活动信息（T-init），而不是时频掩码

（TS-VAD 自然也是如此）。通过 GSS 将其扩展为时频掩码后，该掩码用于声源提取（第（2）行和第（4）行）。两种配置均受益于 GSS，但 TS-VAD 的收益明显大于 TS-SEP，因此二者之间的差距大幅缩小，cpWER 分别为 6.43% 和 6.32%。与 TS-VAD 不同，TS-SEP 可以利用 GSS 进行掩码精炼（TF-init），使 cpWER 略微提升至 6.15%。

对于 GSS，我们使用了 ±15 秒的上下文长度，因此每个应用 EM算法 的片段都会扩展 30 秒。虽然使用 GSS 作为后处理步骤时，TS-SEP 相较于 TS-VAD 的优势似乎几乎消失了，但鉴于 GSS 的计算复杂度很高[12]，这一结论应当重新审视。在图6中，我们考察了基于 TS-SEP 输出的不同初始化方式和不同上下文长度，并使用了至多 20 个引导式和 1 个非引导式 EM 步骤，这些是默认设置。可以看出，GSS 的基于掩码的初始化（TF-init）始终优于基于时间活动状态的初始化（T-init）。此外，使用掩码可以减少 EM 步骤的数量，而不会对 WER 产生显著影响。所需 EM 步骤较少是意料之中的，因为掩码本身已是可靠的起点。

此外，采用掩码初始化后，扩展片段所需的上下文可以显著减少。如果两个说话人在整个片段时长内都重叠，这种歧义也能很好地解释这一点：基于掩码的 TF 级初始化仍能区分说话人，但时间活动信息有时不足以做到这一点，因此需要加入左右上下文，以提高说话人之间的活动模式差异足够大的可能性。有关 GSS 中上下文扩展优势的详细讨论，请参见 [7]。

表 V 还列出了使用 WavLM 模型的结果，由此可以得出类似结论。使用更强大的基于 WavLM 的自动语音识别系统，我们为 TS-SEP 获得了最佳 cpWER，即 5.06%。

请注意，表中还包含 DI-cpWER 结果。该指标与说话人无关，因此不会将说话人分离错误对词错误率的影响计入其中。在此，最佳配置的词错误率为 3.26%。

K. 与文献的比较

LibriCSS 是一个公开的数据集，因此可以将我们的系统与其他研究取得的性能进行比较。我们在表 VI 中汇总了一些相关结果。为求完整，我们还在表VII中报告了系统在不同语音重叠条件下的性能。

在截至目前的实验中，我们遵循 [2] 提出的惯例：将 LibriCSS 的第一个会话用于开发集，其余 9 个用于评估集。对于我们在此报告的部分文献结果，出版物并未明确说明是否遵循了这一惯例，或者这些结果是否

[12]例如，在 CHiME-6 基线系统中， [1], GSS 使用了 12 个麦克风，而非 24 个麦克风，因为运行时间过长。

表 VI 文献比较。风格列表示系统的设计方式，其中 D、S 和 A 分别表示说话人日志、分离和自动语音识别。A 表示序列， + 表示组件之间的耦合。 †表示 cpWER 是在1 分钟数据的10 分钟片段上计算的。

➔

系统（其他）	风格	单一通道	cpWER	SAg-WER asclite
CSS 与到达方向说话人分离 [58]	S ➔ D ➔ A	–	12.98	–
CSS 与到达方向说话人分离 [58]	S ➔ D ➔ A	–	12.40 †	–
SMM [6]	D ➔ S ➔ A	–	5.9 †	–
CSS ➔ 谱聚类 [2]	S ➔ D ➔ A	–	12.7	–
t-SOT TT [59]	E2E	✓	–	7.6
CSS [60]	S ➔ A	✓	–	10.15
CSS [60]	S ➔ A	–	–	5.85
转录到说话人分离 [61]	端到端	✓	11.6	–
TS-VAD + Speakerbeam [12]	D ➔ S ➔ A	✓	18.8	–
TS-VAD + GSS [12]	D ➔ S ➔ A	–	11.2	–
谱聚类 + GSS [62]	D ➔ S ➔ A	–	12.12	–

系统（混合）	风格	单一通道	cpWER	SAg-WER DI-cpWER
谱聚类 [62] ➔ GSS ➔ WavLM	D ➔ S ➔ A	–	13.85	8.82

系统（我们的）	风格	单一通道	cpWER	SAg-WER DI-cpWER
TS-VAD ➔ GSS ➔ Base	D ➔ S ➔ A	–	6.70	4.36
TS-VAD ➔ WavLM	D ➔ A	✓	9.26	7.16
TS-VAD ➔ GSS ➔ WavLM	D ➔ S ➔ A	–	5.77	3.40
TS-SEP ➔ 掩码。➔ Base	D + S ➔ A	✓	11.61	9.66
TS-SEP ➔ GSS ➔ Base	D +S ➔ A	–	6.42	4.34
TS-SEP ➔ 掩码。➔ WavLM	D +S ➔ A	✓	7.81	5.80
TS-SEP ➔ GSS ➔ WavLM	D +S ➔ A	–	5.36	3.27

不同重叠子集的 cpWER 分解：从无重叠且停顿较长（0L）或较短（0S），到重叠比例为 10% 至 40%。

系统	重叠比例（%） 0L	0S	10	20	30	40	avg
TS-VAD ➔ GSS ➔ Base	5.76	4.34	6.73	8.12	8.04	6.53	6.70
TS-VAD ➔ WavLM	5.05	5.04	7.89	10.78	12.27	11.90	9.26
TS-VAD ➔ GSS ➔ WavLM	4.66	3.51	5.96	7.59	7.02	5.24	5.77
TS-SEP ➔ 掩码。➔ Base	7.91	5.57	9.34	13.27	15.09	15.58	11.61
TS-SEP ➔ GSS ➔ 基础版	6.15	3.71	6.32	7.90	7.12	6.80	6.42
TS-SEP ➔ 掩码。➔ WavLM	5.25	3.63	6.63	9.53	9.84	10.10	7.81
TS-SEP ➔ GSS ➔ WavLM	5.02	2.75	5.35	7.14	5.91	5.55	5.36

用于开发集+评估集。为确保公平，我们因此在本表中报告开发集+评估集上的结果；对我们而言，这些结果略差于仅在评估集上的结果。

我们发现，LibriCSS 上最佳的 WER 是 Raj 等人取得的 3.34% [62] 以及 Wang 等人取得的 3.97% [63]。但这两项结果都使用了理想说话人分离。除此之外，据我们所知，Wang 等人 [63]还通过使用多通道，报告了 LibriCSS 上基于 asclite 的最佳 WER，为 5.85%。他们使用了连续语音分离流水线，其中一个系统根据观测生成两条流，每条流均不包含重叠语音。当他们将系统限制为单通道数据时，得到的 WER 为 10.15%。在一项

附带实验中，Kanda 等人 [59][13]，报告了 7.6% 的基于 asclite 的 WER；他们的系统仅使用单一通道，并采用序列化输出训练，该系统可直接生成转录文本，而不是先生成中间的无重叠音频信号。随后，Kanda 等人 [61] 提出了转录到说话人分离，其中包含说话人分离信息，并报告了 11.6% 的 cpWER，这是我们发现的最佳单通道 cpWER。

在 [12], Delcroix 等人报告了将 TS-VAD 与 Speakerbeam 结合用于单通道处理，并获得了 18.8% 的 cpWER。为了解释这一 WER，他们还尝试了将 TS-VAD 与 GSS 结合，并获得了 11.2% 的 cpWER，这是我们在 LibriCSS 上找到的最佳 cpWER（不受单通道约束）。

在 [6], LibriCSS 上应用了 SMM，但其内存消耗存在问题，因此仅应用于平均时长约为 1 分钟的子分段数据；这类数据通常用于采用 asclite 进行 WER 评估的连续语音分离流水线。SMM 在子分段数据上获得了 5.9% 的 cpWER。在 [58], 基于连续语音分离的分离之后，使用到达方向（DOA）方法进行说话人日志处理。这是少数同时报告 1 分钟数据和完整 10 分钟迷你会话 cpWER 的出版物之一，因此可用于估计会话时长对 WER 性能的影响。他们观察到，使用较短数据时性能提升了 0.58%。由于较短的数据可以补偿 cpWER 中的一些说话人分离错误，因此出现性能提升是合理的。

使用2020年年中公开提供、适用于所有提及的参考系统并用于 [6], 的预训练“base”自动语音识别系统，我们的系统能够优于或达到与所有系统相当的表现，但在单通道约束下，基于 asclite 的词错误率除外 [59] 。使用近期发布的基于“WavLM”的自动语音识别系统，我们能够优于所有系统。值得注意的是，我们重新实现的 TS-VAD 与预训练的“WavLM”结合使用，且不进行任何增强时，构成一个单通道系统；在 10 分钟数据上的 cpWER 优于所有参考系统，无论它们使用单通道还是多通道数据。

[62] 的说话人分离估计可公开获取[14]。我们对其应用了 GSS 和基于“WavLM”的自动语音识别系统，观察到 cpWER有所下降，这凸显了我们的说话人分离性能之强。

有趣的是，在解决更困难任务的同时，我们得到的一些词错误率也优于文献中的结果。即使不使用理想说话人分离，我们也获得了 3.27% 的说话人无关词错误率，而迄今为止最佳词错误率为 3.34% [62], ，该结果使用了理想说话人分离。虽然 [63]使用说话人无关词错误率得到 5.85%，但我们提出的系统能够达到 5.36% 的词错误率，同时还将说话人分离造成的错误计入其中。

VI. 结论与展望

通过将输出从时间分辨率扩展到时频分辨率，我们得以将 TS-VAD 说话人日志系统扩展为一种联合说话人日志与分离方法。我们将这种方法命名为 TS-SEP；它具有通用性，可用于单通道和多通道输入数据。TS-SEP 可被视为 GSS 的替代方案，也可以与 GSS 结合，在多通道数据上进一步提升性能。我们讨论了 TS-VAD 和 TS-SEP 的各种训练细节，并在 LibriCSS 会议数据集上进行了全面的实验评估。我们通过说话人日志错误率，以及说话人日志相关和无关的词错误率来衡量性能，以评估说话人分离错误对自动语音识别的影响。采用“基础版”自动语音识别模型时，单通道和多通道情况下的词错误率分别为 11.61% 和 6.42%；使用 WavLM 特征进行自动语音识别时，分别为 7.81% 和 5.36%。据此，我们在说话人日志相关词错误率方面达到了新的最先进水平。最佳结果在说话人日志错误无关的评估中对应的词错误率为 3.27%。

今后，我们计划将 TS-SEP 应用于 CHiME-6 等真实录音。由于 TS-SEP 也可以采用 TS-VAD 目标进行训练，因此可以进行混合训练：在真实数据上同时使用语音活动检测目标进行训练，并在模拟数据上使用分离目标进行训练，从而实现在没有分离目标的情况下，利用真实数据进行分离训练。

[13]详见 arXiv 版本的附录：https://arxiv.org/abs/220200842 14 。 https://github.com/desh2608/diarizer

致谢

本工作的部分内容是在 Christoph Boeddeker 于三菱电机研究实验室（MERL）实习期间完成的。之后，他获得了德国研究基金会（DFG）的资助，项目编号为 448568305。帕德博恩并行计算中心提供了计算资源。

参考文献

[1] S. Watanabe、M. Mandel、J. Barker、E. Vincent、A. Arora、X. Chang、S. Khudanpur、V. Manohar、D. Povey、D. Raj、D. Snyder、A. S. Subrama- nian、J. Trmal、B. B. Yair、C. Boeddeker、Z. Ni、Y. Fujita、S. Horiguchi、N. Kanda、T. Yoshioka 和 N. Ryant，《CHiME-6 挑战赛：应对未分段录音中的多说话人语音识别》，载于*CHiME-6*研讨会论文集，2020年。[2] D. Raj、P. Denisov、Z. Chen、H. Erdogan、Z. Huang、M. He、S. Watanabe、J. Du、T. Yoshioka、Y. Luo、N. Kanda、J. Li、S. Wisdom 和 J. R. Her- shey，《面向多说话人会议的语音分离、说话人日志与识别集成：系统描述、比较与分析》，载于*IEEE SLT*论文集，2021年，第897–904页。[3] Z. Chen、T. Yoshioka、L. Lu、T. Zhou、Z. Meng、Y. Luo、J. Wu、X. Xiao 和 J. Li，《连续语音分离：数据集与分析》，载于*IEEE ICASSP*论文集，2020年，第7284–7288页。[4] N. Ryant、K. Church、C. Cieri、A. Cristia、J. Du、S. Ganapathy 和 M. Liberman，《第二届 DIHARD 说话人日志挑战赛：数据集、任务与基线系统》，载于*ISCA Interspeech*论文集，2019年，第978–982页。[5] N. Ito、S. Araki 和 T. Nakatani，《基于频率无关声源存在先验的全频带聚类实现无置换卷积盲源分离》，载于*IEEE ICASSP*论文集，2013年，第3238–3242页。[6] C. Boeddeker、T. Cord-Landwehr、T. von Neumann 和 R. Haeb- Umbach，《基于空间混合模型的会议分离初始化方案》，载于*ISCA Interspeech*论文集，2022年，第271–275页。[7] C. Boeddeker、J. Heitkaemper、J. Schmalenstroeer、L. Drude、J. Hey- mann 和 R. Haeb-Umbach，《CHiME-5 晚宴场景的前端处理》，载于*CHiME-5*研讨会论文集，2018年。[8] I. Medennikov、M. Korenevsky、T. Prisyach、Y. Khokhlov、M. Ko- renevskaya、I. Sorokin、T. Timofeeva、A. Mitrofanov、A. Andrusenko、I. Podluzhny、A. Laptev 和 A. Romanenko，《面向 CHiME-6 挑战赛的 STC 系统》，载于*CHiME-6*研讨会论文集，2020年，第36–41页。

[9] A. Eisenberg、B. Schwartz 和 S. Gannot，《使用递归期望最大化的在线盲音频源分离》，载于 ISCA Interspeech论文集，2021年，第3480–3484页。[10] K. ˇZmol´ıkov´a、M. Delcroix、K. Kinoshita、T. Higuchi、A. Ogawa 和 T. Nakatani，《用于语音混合信号中说话人提取的说话人感知神经网络波束形成器》，载于 ISCA Interspeech论文集，2017年，第2655–2659页。[11] Q. Wang、H. Muckenhirn、K. Wilson、P. Sridhar、Z. Wu、J. R. Hershey、R. A. Saurous、R. J. Weiss、Y. Jia 和 I. L. Moreno，《Voicefilter：通过说话人条件频谱图掩蔽实现目标语音分离》，载于 ISCA Interspeech论文集，2019年，第2728–2732页。[12] M. Delcroix、K. Zmolikova、T. Ochiai、K. Kinoshita 和 T. Nakatani，《说话人活动驱动的神经语音提取》，载于 IEEE ICASSP论文集，2021年，第6099–6103页。[13] Y. Fujita、N. Kanda、S. Horiguchi、K. Nagamatsu 和 S. Watanabe，《采用无置换目标函数的端到端神经说话人日志》，载于 ISCA Interspeech 论文集，2019 年，第 4300–4304 页。[14] S. Horiguchi、Y. Fujita、S. Watanabe、Y. Xue 和 K. Nagamatsu，《采用基于编码器-解码器的吸引子的未知说话人数端到端说话人日志》，载于 ISCA Interspeech论文集，2020年，第269–273页。[15] S. Horiguchi、S. Watanabe、P. Garc´ıa、Y. Xue、Y. Takashima 和 Y. Kawaguchi，《利用全局与局部吸引子实现不限说话人数的神经说话人日志》，载于 IEEE ASRU论文集，2021年，第98–105页。[16] K. Kinoshita、T. von Neumann、M. Delcroix、C. Boeddeker 和 R. Haeb-Umbach，《采用 Graph-PIT 的逐话语重叠感知神经说话人日志》，载于 ISCA Interspeech 论文集，2022年，第1486–1490页。[17] K. Kinoshita、M. Delcroix 和 N. Tawara，《端到端神经说话人日志与基于聚类的说话人日志集成：兼得两者之长》，载于 IEEE ICASSP论文集，2021年，第7198–7202页。[18] I. Medennikov、M. Korenevsky、T. Prisyach、Y. Khokhlov、M. Korenevskaya、I. Sorokin、T. Timofeeva、A. Mitrofanov、A. Andrusenko、I. Podluzhny、A. Laptev 和 A. Romanenko，《目标说话人语音活动检测：晚宴场景中多说话人日志的一种新方法》，载于 ISCA Interspeech论文集，2020年，第274–278页。[19] S. Ding、Q. Wang、S.-Y. Chang、L. Wan 和 I.-L. Moreno，《个人语音活动检测：说话人条件语音活动检测》，载于 Odyssey说话人与语言识别研讨会论文集，2020年，第433–439页。[20] M. He、D. Raj、Z. Huang、J. Du、Z. Chen 和 S. Watanabe，《通过改进未知说话人数情况下的 i-vector 估计实现目标说话人语音活动检测》，载于 ISCA Interspeech论文集，2021年，第3555–3559页。[21] D. Wang、X. Xiao、N. Kanda、T. Yoshioka 和 J. Wu，《基于 Transformer 的目标说话人语音活动检测及其与端到端神经说话人日志的集成》，载于 IEEE ICASSP论文集。IEEE，2023年。[22] X. Xiao、Z. Chen、T. Yoshioka、H. Erdogan、C. Liu、D. Dimitriadis、J. Droppo 和 Y. Gong，《使用说话人库和注意力网络进行单通道语音提取》，载于 IEEE ICASSP论文集，2019年，第86–90页。[23] N. Zeghidour 和 D. Grangier，《Wavesplit：通过说话人聚类实现端到端语音分离》，《IEEE/ACM音频、语音与语言处理汇刊》，第29卷，第2840–2849页，2021年。[24] S. Maiti、Y. Ueda、S. Watanabe、C. Zhang、M. Yu、S.-X. Zhang 和 Y. Xu，《EEND-SS：适用于灵活说话人数的端到端神经说话人日志与语音分离联合方法》，载于 IEEE SLT论文集，2023年，第480–487页。[25] D. Snyder、D. Garcia-Romero、G. Sell、D. Povey 和 S. Khudanpur，《X向量：用于说话人识别的鲁棒 DNN嵌入》，载于 IEEE ICASSP论文集，2018年，第5329–5333页。[26] T. J. Park、K. J. Han、M. Kumar 和 S. Narayanan，《使用归一化最大特征间隙自动调节说话人日志的谱聚类》，《IEEE信号处理快报》，第27卷，第381–385页，2019年。[27] N. Dehak、P. J. Kenny、R. Dehak、P. Dumouchel 和 P. Ouellet，《用于说话人验证的前端因子分析》，《IEEE音频、语音与语言处理汇刊》，第19卷，第4期，第788–798页，2010年。[28] Z.-Q. Wang、J. Le Roux 和 J. R. Hershey，《多通道深度聚类：用于说话人无关语音分离的判别性频谱与空间嵌入》，载于 IEEE ICASSP论文集，2018年，第1–5页。[29] J. Heymann、L. Drude 和 R. Haeb-Umbach，《用于声学波束形成的神经网络频谱掩码估计》，载于 IEEE ICASSP论文集，2016年，第196–200页。[30] J. Heitkaemper、D. Jakobeit、C. Boeddeker、L. Drude 和 R. Haeb-Umbach，《揭开 TasNet 的神秘面纱：一种剖析式方法》，载于 IEEE ICASSP论文集，2020年，第6359–6363页。

[31] T. Cord-Landwehr、C. Boeddeker、T. von Neumann、C. Zorila、R. Doddipatla 和 R. Haeb-Umbach，“单通道干源分离：从无混响环境到混响环境”，载于 IEEE IWAENC 会议论文集，2022年。[32] T. von Neumann、K. Kinoshita、C. Boeddeker、M. Delcroix 和 R. Haeb-Umbach，“SA-SDR：一种用于会议风格数据分离的新型损失函数”，载于 IEEE ICASSP 会议论文集，2022年，第 6022–6026 页。[33] R. M. Haralick、S. R. Sternberg 和 X. Zhuang，“使用数学形态学进行图像分析”，《IEEE 模式分析与机器智能汇刊》，第 4 期，第 532–550 页，1987年。[34] M. Souden、J. Benesty 和 S. Affes，“关于用于降噪的最优频域多通道线性滤波”，《IEEE 音频、语音与语言处理汇刊》，第 18 卷，第 2 期，第 260–276 页，2009年。[35] A. S. Subramanian、C. Weng、M. Yu、S.-X. Zhang、Y. Xu、S. Watanabe 和 D. Yu，“使用端到端语音识别目标进行远场位置引导的目标语音提取”，载于 IEEE ICASSP 会议论文集，2020年，第 7299–7303 页。[36] K. Iwamoto、T. Ochiai、M. Delcroix、R. Ikeshita、H. Sato、S. Araki 和 S. Katagiri，“伪影有多严重？：分析语音增强错误对自动语音识别的影响”，载于 Interspeech 2022 会议论文集，2022年，第 5418–5422 页。[37] J. Barker、S. Watanabe、E. Vincent 和 J. Trmal，“第五届 ‘CHiME’ 语音分离与识别挑战赛：数据集、任务和基线系统”，载于 ISCA Interspeech 会议论文集，2018年。[38] O. Sadjadi、C. Greenberg、E. Singer、L. Mason 和 D. Reynolds，“美国国家标准与技术研究院 2021 年说话人识别评测计划”，2021年。[39] T. von Neumann、C. Boeddeker、K. Kinoshita、M. Delcroix 和 R. Haeb-Umbach，“关于多说话人语音识别系统中词错误率的定义及其高效计算”，载于 IEEE ICASSP 会议论文集，2023年。[40] J. G. Fiscus、J. Ajot、N. Radde 和 C. Laprun，“用于在同时语音场景下评估自动语音识别系统的多维莱文斯坦编辑距离计算”，载于 LREC 会议论文集，2006年，第 803–808 页。[41] S. Watanabe、T. Hori、S. Karita、T. Hayashi、J. Nishitoba、Y. Unno、N. Enrique Yalta Soplin、J. Heymann、M. Wiesner、N. Chen、A. Renduch- intala 和 T. Ochiai，“ESPnet：端到端语音处理工具包”，载于 ISCA Interspeech 会议论文集，2018年，第 2207–2211 页。[42] S. Watanabe，“ESPnet2 预训练模型，Shinji Watanabe/librispe ech asr train asr transformer e18 raw bpe sp valid .acc.best，fs=16k，lang=en”，2020年。 [在线]。获取地址：https://doi.org/10.5281/zenodo.3966501[43] X. Chang、T. Maekaku、Y. Fujita 和 S. Watanabe，“语音识别、语音增强与自监督表征学习的端到端集成”，载于 ISCA Interspeech 会议论文集，2022年，第 3819–3823 页。[44] S. Chen、C. Wang、Z. Chen、Y. Wu、S. Liu、Z. Chen、J. Li、N. Kanda、T. Yoshioka、X. Xiao、J. Wu、L. Zhou、S. Ren、Y. Qian、Y. Qian、J. Wu、M. Zeng、X. Yu 和 F. Wei，“WavLM：面向全栈语音处理的大规模自监督预训练”，《IEEE 信号处理精选主题汇刊》，第 16 卷，第 6 期，第 1505–1518 页，2022年。[45] E. Vincent、S. Watanabe、A. A. Nugraha、J. Barker 和 R. Marxer，“鲁棒语音识别中的环境、麦克风和数据仿真失配分析”，《计算机语音与语言》，第 46 卷，第 557 页，2017年 535pp. – 。[46] J. Kahn、M. Riviere、W. Zheng、E. Kharitonov、Q. Xu、P.-E. Mazar´e、J. Karadayi、V. Liptchinsky、R. Collobert、C. Fuegen、T. Likhomanenko、G. Synnaeve、A. Joulin、A.-r. Mohamed 和 E. Dupoux，“Libri-Light：面向有限或无监督自动语音识别的基准”，载于 IEEE ICASSP 会议论文集，2020年，第 7669–7673 页。[47] G. Chen、S. Chai、G.-B. Wang、J. Du、W.-Q. Zhang、C. Weng、D. Su、D. Povey、J. Trmal、J. Zhang、M. Jin、S. Khudanpur、S. Watanabe、S. Zhao、W. Zou、X. Li、X. Yao、Y. Wang、Z. You 和 Z. Yan，“GigaSpeech：不断发展的多领域自动语音识别语料库，包含 10,000 小时转录音频”，载于 ISCA Interspeech 会议论文集，2021年，第 3670–3674 页。[48] C. Wang、M. Riviere、A. Lee、A. Wu、C. Talnikar、D. Haziza、M. Williamson、J. Pino 和 E. Dupoux，“VoxPopuli：用于表征学习、半监督学习和解释的大规模多语言语音语料库”，载于 ACL 会议论文集。计算语言学协会，2021年 8 月，第 993–1003 页。[49] D. B. Paul 和 J. Baker，“基于《华尔街日报》的连续语音识别语料库设计”，载于 DARPA语音与自然语言研讨会论文集，1992年。[50] D. Povey、A. Ghoshal、G. Boulianne、L. Burget、O. Glembek、N. Goel、M. Hannemann、P. Motlicek、Y. Qian、P. Schwarz、J. Silovsky、G. Stem- mer 和 K. Vesely，“Kaldi 语音识别工具包”，载于 IEEE ASRU，2011.

[51] T. Nakatani、T. Yoshioka、K. Kinoshita、M. Miyoshi 和 B.-H. Juang，“基于方差归一化延迟线性预测的语音去混响”，*IEEE*《音频、语音与语言处理汇刊》，第 18 卷，第 7 期，第 1717–1731 页，2010 年。[52] L. Drude、J. Heymann、C. Boeddeker 和 R. Haeb-Umbach，“NARA-WPE：用于 NumPy 和 TensorFlow 的加权预测误差去混响 Python 软件包，支持在线和离线处理”，载于《语音通信；第 *13* 届 *ITG* 研讨会》。VDE，2018年，第 1–5 页。[53] J. Ebbers 和 R. Haeb-Umbach，“家庭环境中的自训练音频标注与声音事件检测”，载于*DCASE* 会议论文集，2021年。[54] H. Zhang、M. Cisse、Y. N. Dauphin 和 D. Lopez-Paz，“Mixup：超越经验风险最小化”，载于*ICLR* 会议论文集，2018年。[55] A. Paszke、S. Gross、F. Massa、A. Lerer、J. Bradbury、G. Chanan、T. Killeen、Z. Lin、N. Gimelshein、L. Antiga、A. Desmaison、A. Kopf、E. Yang、Z. DeVito、M. Raison、A. Tejani、S. Chilamkurthy、B. Steiner、L. Fang、J. Bai 和 S. Chintala，“PyTorch：一种命令式风格的高性能深度学习库”，载于《神经信息处理系统进展》第 *32* 卷。Curran Associates, Inc.，2019年，第 8024–8035 页。[56] S. Hochreiter 和 J. Schmidhuber，“长短期记忆”，《神经计算》，第 9 卷，第 8 期，第 1735–1780 页，1997 年。[57] H. Sak、A. Senior 和 F. Beaufays，“用于大词汇量语音识别的长短期记忆循环神经网络架构”，*arXiv* 预印本 *arXiv:1402.1128*，2014 年。[58] Z.-Q. Wang 和 D. Wang，“基于定位的连续语音分离序列分组”，载于*IEEE ICASSP* 会议论文集，2022年，第 281–285 页。[59] N. Kanda、J. Wu、Y. Wu、X. Xiao、Z. Meng、X. Wang、Y. Gaur、Z. Chen、J. Li 和 T. Yoshioka，“采用词元级序列化输出训练的流式多说话人自动语音识别”，载于*ISCA Interspeech* 会议论文集，2022年，第 3774–3778 页。[60] Z.-Q. Wang、P. Wang 和 D. Wang，“用于逐句及连续语音分离的多麦克风复谱映射”，*IEEE/ACM*《音频、语音与语言处理汇刊》，第 29 卷，第 2001–2014 页，2021年。[61] N. Kanda、X. Xiao、Y. Gaur、X. Wang、Z. Meng、Z. Chen 和 T. Yoshioka，“转录到说话人分离：利用端到端说话人归属自动语音识别实现不限说话人数的神经说话人分离”，载于*IEEE ICASSP* 会议论文集，2022年，第 8082–8086 页。[62] D. Raj、D. Povey 和 S. Khudanpur，“用于会议转录的 GPU加速引导源分离”，载于*ISCA Interspeech* 会议论文集，2023年，第 3507–3511 页。[63] Z.-Q. Wang、P. Wang 和 D. Wang，“用于逐句及连续语音分离的多麦克风复谱映射”，*IEEE/ACM*《音频、语音与语言处理汇刊》，第 29 卷，第 2001–2014 页，2021年。