用于说话人验证的 Koopman 正则化深度语音解耦

Nikos Chazaridis，*IEEE* 学生会员，Mohammad Belal，Rafael Mestre，Timothy J. Norman *IEEE* 会员，Christine Evers，*IEEE* 高级会员

arXiv:2603.05577v1 [cs.SD] 5 Mar 2026

摘要—**人类语音同时包含语言内容和说话人相关特征，因此说话人验证是身份关键型应用中的一项关键技术。现代深度学习说话人验证系统旨在学习不受语义内容和环境噪声等干扰因素影响的说话人表征。然而，许多现有方法依赖标注数据、文本监督或大型预训练模型作为特征提取器，这限制了其可扩展性和实际部署，并引发了可持续性方面的担忧。我们提出了深度库普曼语音解耦自编码器（DKSD-AE），这是一种结构化自编码器，将一种新颖的多步 Koopman 算子学习模块与实例归一化相结合，以解耦说话人和内容动态。在多个数据集上开展的定量实验表明，与最先进的基线方法相比，DKSD-AE 的说话人验证性能有所提升或具有竞争力，同时保持较高的内容等错误率，证实了其能够有效解耦。这些结果是在参数量大幅减少且无需文本监督的情况下取得的。此外，在评估规模扩大时，模型性能仍保持稳定，凸显了表征的鲁棒性和泛化能力。我们的研究结果表明，将基于库普曼的时序建模与实例归一化相结合，可为面向说话人的表征学习提供一种高效且有理有据的解决方案。**

索引词—**说话人验证、解纠缠、Koopman 算子理论。**

I. 引言

语音信号传递多层信息，包括语言内容、表达线索和说话人特定特征。说话人验证（说话人验证）利用这些特征来区分合法用户与冒充者。高效的说话人验证对于安全的远程服务至关重要，例如金融交易中的语音身份认证。在这些应用中，系统必须实现高准确率，同时满足计算效率方面的约束。因此，需要学习能够将说话人特定信息与其他语音属性分离的表示。

当代基于深度学习的说话人验证系统包含两个阶段。首先是注册阶段，从目标话语中提取说话人表征；随后是评估阶段，将未见过的话语与已注册的说话人进行比较。经典的说话人验证系统利用通过卷积神经网络（CNNs）或其变体生成的说话人表征 [1], [2]。近年来，大型预训练语音表示模型（如 HuBERT [3] 和 WavLM [4] ）也被用作说话人验证系统的特征提取器，既可与原始声学特征结合使用，也可作为独立输入 [5], [6]。

然而，现有的表示方法需要大规模训练语料库。经典的说话人验证系统需要大量人工标注，而基于大型预训练语音模型的现代方法则依赖数千小时的数据以及大量计算资源。尽管有效，这一范式资源消耗巨大，并引发可持续性方面的担忧，尤其是当宽泛的通用模型被用于说话人验证这类定义狭窄的任务时。

语音信号中的某些信息源与说话人验证无关 [7]。学习仅保留说话人特有特征并抑制无关因子的表示，对于有效的说话人验证至关重要。这一观察促使我们学习能够明确将说话人特定信息与其他语音属性分离的结构化表征。因此，本文的目标是提出一种新颖的语音表征学习方法，使其能够在不依赖说话人标签或大型基础模型的情况下，有效地将说话人身份与其他语音属性分解开来，从而避免相应的标注成本与计算成本。

为了从语音数据中分离出感兴趣的信息源，我们采用解耦表示学习。解纠缠旨在使用相互分离的潜在因子来表示数据，这些潜在因子对应于语义上有意义的变异来源 [8]。在实践中，此类表示通常使用自编码器或变分自编码器（VAEs） [9]来学习。然而，在没有额外约束的情况下，无监督的解耦表示学习从根本上是不适定的，因为所学表示对超参数选择和初始化高度敏感，从而导致不一致的结果以及较差的泛化能力 [10]。因此，有意义的解纠缠需要引入明确的假设（通常称为归纳偏置），以限制可接受解的空间并引导学习。尽管颇具吸引力，基于变分自编码器的解耦方法仍容易出现后验坍塌，因为训练目标可能对潜在空间过度正则化，同时允许表达能力强的解码器忽略潜在表示。这会导致模型无法解耦有意义的变异因子 [11]。

语音信号也呈现多尺度时间结构。与先前的工作 [12], [13], 一样，我们假设在一个话语中，说话人身份的变化比语言内容更缓慢。基于这一假设，我们将准静态

本文已提交 IEEE，供其考虑发表。版权可能在未另行通知的情况下转让，届时本文版本可能不再可访问。

N.Chazaridis、Mohammad Belal、Rafael Mestre、Timothy J.Norman 和 Christine Evers 就职于英国南安普顿大学电子与计算机科学学院，邮编 SO171BJ，南安普顿，英国（电子邮箱：n.chazaridis@soton.ac.uk；r.mestre@soton.ac.uk；t.j.norman@soton.ac.uk；c.evers@soton.ac.uk）

Mohammad Belal 也任职于英国南安普敦 SO14 3ZH 的国家海洋学中心海洋地球科学部（电子邮箱：mob@noc.ac.uk）

语音属性建模为缓慢的系统动力学，并通过架构和算子理论归纳偏置来实现这一点。Koopman 算子理论 [14] 为我们提供了一种实现这一目标的方法，因为它能够通过一个线性但无限维的算子来建模系统动力学，该算子控制潜变量的演化。这种表述使时间行为能够通过对算子施加谱约束或结构约束而得到明确表示和塑造 [15]。

对于说话人验证，语言内容是一种干扰变量，因此有效的说话人表征应不受言语内容变化的影响。为明确抑制依赖于内容和信道的统计特性，我们将实例归一化[16] 作为归纳偏置，与 Koopman 算子学习相结合。研究表明，实例归一化有助于在语音转换 [17], [18],等语音处理任务中分离说话人相关特征，但仅靠它无法保证说话人身份与语音内容的解纠缠。

在这项工作中，我们提出了一种新颖的自编码器，通过两个专用编码器分支，将语音频谱图解纠缠为说话人表征和内容表征。我们将实例归一化与多步 Koopman 算子学习模块相结合，以实现潜在空间的时间分离：实例归一化用于捕捉快速变化的语音内容动态，而多步 Koopman 算子学习模块则用于建模缓慢演变的说话人相关属性。多步 Koopman 算子学习使动力学编码器能够捕捉长期时间结构，从而提取细粒度的说话人表征；与之互补的内容分支则专注于与身份无关的变化。我们的贡献如下：

1) **通过时间归纳偏置实现结构化解纠缠**。我们提出深度 Koopman 说话人解纠缠自编码器（DKSD-AE），这是一种双分支架构，可将快速变化的内容动态与缓慢演化的说话人特征分离。实例归一化有助于提升对话语级变异的鲁棒性，而正则化Koopman算子学习则对说话人属性的结构化时间演化进行建模。

2)**用于长期依赖建模和表示稳定性的多步 Koopman 算子学习**。我们提出一种新颖的多步预测形式，通过近似 Koopman 算子，对高维语音数据中的长程动态进行建模。消融研究表明，与单步神经Koopman方法以及仅采用实例归一化的配置相比，这种形式能够提升说话人验证性能。

3)**准确、高效且可扩展的说话人验证**。DKSD-AE 在 VCTK 上的说话人等错误率（EER）低于所有基线方法，在 TIMIT 上则低于除一种以外的所有基线方法，同时取得较高的内容等错误率，证实了其有效的解纠缠能力。与基线方法相比，该方法使用的参数显著更少，且无需文本监督，仅依赖梅尔频谱图输入即可取得这些结果。最后，鲁棒性测试表明，将测试集规模从 TIMIT 官方测试集扩大到近七倍大的 TIMIT 完整版测试集时，EER 仅下降 $\approx 1\%$。

组，进一步凸显了表示稳定性。

本文其余部分安排如下。第二节回顾与本研究相关的工作。第三节介绍 Koopman 算子模块所需的理论预备知识。第四节介绍 DKSD-AE 框架。第五节和第六节分别介绍实验设置和结果。第七节总结全文。

II. 相关工作

A. 解耦说话人表征

序列到序列自编码器网络因其架构能够将多维时间序列投影到较低维空间，并从中分离出关注的特征，因此被广泛用于解纠缠。近期语音模型会将话语解耦到不同的潜在空间，以表示说话人身份、语言内容和情感等因素 [12], [19]–[21]。

DSVAE 等无监督序列到序列概率方法及其扩展 [5], [13], [22] 通过对两个独立的潜在空间施加维度约束来实现解纠缠：说话人信息由固定大小的潜在向量表示，而内容则由潜变量序列捕获。DSVAE 架构经过扩展，通过在损失函数中加入互信息项来利用对比学习，从而促进静态与动态潜变量之间的独立性 [23]。此外，一些研究采用多分支编码器，将语音划分为不同的潜在表示。例如，SpeechTripleNet [20] 采用三分支变分自编码器，每个分支都以特定模态为条件，以解纠缠内容、音色和韵律。为引导解纠缠，相关方法会将辅助模态与原始语音结合使用。例如，VAE-TP [24],以文本标注作为内容分支的条件，强制将语言信息与说话人特定因素分离。尽管这些方法效果显著，但它们依赖于训练期间可用的同步元数据。我们的模型受到这类序列到序列方法的启发：这些方法通过明确的架构和训练偏置来实现解纠缠，但不需要文本标注或说话人标签。

B. 算子理论表征

近年来，越来越多的方法开始研究如何利用算子理论表示来学习序列数据的表征。具体而言，许多方法借助 Koopman算子，通过用一个无限维但线性的算子对非线性系统$\mathbf{F}$ 的状态转移进行建模，从而将其线性化 $\mathcal{K}$ [14]。其动机在于，与原始非线性系统相比，线性替代模型$\mathbf{F}$ 可简化后续分析。尽管 $\mathcal{K}$ 其维度是无限的，实际应用中仍希望得到一个有限维近似$\mathbf{K}$ ，以充分编码$\mathbf{F}$中的动力学。动态模态分解（DMD） [25] 及其变体（如扩展动态模态分解（eDMD） [26] ）是用于估计 $\mathcal{K}$ 的代表性方法，但在高维场景下可能存在数值不稳定且对噪声敏感的问题 [27], [28]。

不过，也可以借助神经网络（即自编码器）来估计 $\mathcal{K}$ [29], [30]。神经 Koopman 算子表示主要用于序列数据预测 [31]–[33] ，而对解耦表示学习 [34], [35] 或语音应用的探索较少。本文着重对语音序列中的长期动力学进行建模，并将内容和说话人属性解耦为不同的表示。我们的方法与序贯 Koopman 解耦（SKD）模型[34] 相关；该模型通过对单个 Koopman算子$\mathbf{K}$施加谱约束，将时间序列数据解耦为静态表示和动态表示，该算子用于建模完整的系统动力学。与 [34]，不同，我们的方法采用结构化分解。Koopman 算子学习仅应用于准静态的说话人相关子空间，并通过专门的编码器分支实现；而变化较快的语音成分则由 DKSD-AE 框架中的实例归一化和另一个编码器分支分别处理。像 SKD 那样用单个矩阵表示高维信号的多尺度动力学，会对模型施加相当大的限制，尤其是考虑到近期机器学习方法依赖于表达能力更强的架构。这一观察与先前研究相符：有研究将 Koopman 嵌入用作特征提取步骤，以训练独立的预测模型 [32] ；另有研究采用多个 Koopman 算子来捕捉多尺度动力学，以实现解耦 [35] 和预测 [36]。

III. 预备知识

A. Koopman 算子理论

许多时变现象都可以通过引入状态$\mathbf{x}_t$ 以及状态空间 $\mathcal{X}$上的一般非线性映射 $\mathbf{F}$ ，表示为动力系统：

$$\mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t) \tag{1}$$

在非线性情况下，系统分析可能较为困难，因为非线性系统的某些实现对初始条件敏感，起始值的微小变化就可能导致不同结果；此外，还可能缺乏全局解，以及存在其他难以处理的问题。作为替代方案，我们可以寻找$\mathbf{F}$的线性替代模型，从而利用线性动力系统现有的谱工具揭示其动力学 [14]。

Koopman 算子理论 [14]通过将关注点从状态$\mathbf{x}_t$ 转移到可观测函数 $g : \mathcal{X} \to \mathbb{C}$，提供了这样一种替代方案。Koopman算子 $\mathcal{K}$ 通过复合运算对可观测函数进行线性作用。对于离散时间系统，我们有：

$$(\mathcal{K}g)(\mathbf{x}_t) = g(\mathbf{F}(\mathbf{x}_t)) = g(\mathbf{x}_{t+1}), \tag{2}$$

从而在函数空间 [28] 中形成对非线性动力学的线性但无限维的表示。将非线性动力学约化为线性空间后，即可通过对 Koopman算子进行谱分解来开展系统分析。然而，由于 Koopman算子是无限维的，其计算和分析存在实际问题。为解决这一局限， [29], [30] 利用自编码器生成的有限维表示来近似 Koopman算子。通过多个编码器层学习非线性可观测函数，自编码器能够生成具有表达力且低维的潜在表示，并可在其中通过最小二乘法高效估计Koopman算子。

B. 有限维 Koopman 近似的自编码器

一种常见的、利用神经网络估计 Koopman算子 $\mathcal{K}$ 的数据驱动方法，是设计一个带有瓶颈层的自编码器网络，用于估计有限维近似$\mathbf{K}$ [29]。在一般的 Koopman 自编码器框架中，编码器将状态$\mathbf{x}_t$ 非线性提升到可观测函数空间中，同时约束潜在动力学以线性方式演化，这与 eDMD 类似。扩展动态模态分解通过对状态的一组有限非线性可观测函数的演化进行建模，实现对$\mathbf{K}$ 的估计 [26]。实际上，eDMD 依赖手工设计的可观测函数，在缺乏系统动力学先验知识的情况下，很难选择合适的字典。相比之下，Koopman 自编码器直接从数据中学习这些可观测函数。编码器的输出$\mathbf{Z}$ 用于计算$\mathbf{K}$。

与 eDMD公式 [26], [30], 类似，Koopman自编码器瓶颈 [29], [30], 中的常见做法是，将潜在动力学张量 $\mathbf{Z} \in \mathbb{R}^{T\times d}$ 拆分为两个重叠的快照，其差值为 Δt ($\Delta t = 1$ 在此处)：

$$\mathbf{Z}^- = \begin{bmatrix} | & | & & | \\ \mathbf{z}(t_0) & \mathbf{z}(t_1) & \dots & \mathbf{z}(t_{T-1}) \\ | & | & & | \end{bmatrix}, \tag{3}$$

$$\mathbf{Z}^+ = \begin{bmatrix} | & | & & | \\ \mathbf{z}(t_1) & \mathbf{z}(t_2) & \dots & \mathbf{z}(t_T) \\ | & | & & | \end{bmatrix} \tag{4}$$

然后，通过求解以下回归得到最佳拟合的Koopman算子估计$\mathbf{K}$：

$$\mathbf{K} = \arg\min_{\mathbf{K}} \left\| \mathbf{Z}^+ - \mathbf{Z}^-\mathbf{K} \right\|_F = \mathbf{Z}^+(\mathbf{Z}^-)^\dagger, \tag{5}$$

其中 $\|\cdot\|_F$ 表示Frobenius范数，而$(\mathbf{Z}^-)^\dagger$ 表示$\mathbf{Z}^-$的 Moore-Penrose伪逆。通常，Koopman自编码器通过采用真实与预测潜在动力学张量 $\hat{\mathbf{Z}}^+$的均方误差损失来指导$\mathbf{K}$ 的学习。

IV. 提出的方法

我们提出的方法建立在 Koopman 自编码器 [29], [30] 和实例归一化 [17] 的研究进展之上，旨在将语音中的说话人属性和内容属性解耦为不同的表示。我们提出了 DKSD-AE，这是一种新颖的自编码器神经网络架构，采用双分支编码器和单个解码器 q_{dec}。令$\mathbf{X} \in \mathbb{R}^{B\times T\times F}$ 表示一个三维输入频谱图数据张量，其中B 为训练批次大小维度， T 为频谱图时间帧数， F 为其频率箱数。动力学编码器 f_{dyn} 通过学习频谱约束的Koopman算子，将语音中的准静态属性编码为说话人身份表示$\mathbf{Z}_{\text{s}} \in \mathbb{R}^{B\times T\times k}$。我们采用一种专为编码长期依赖关系而设计的新型多步预测公式，从而实现对 $\mathbf{K}$ Koopman算子的学习

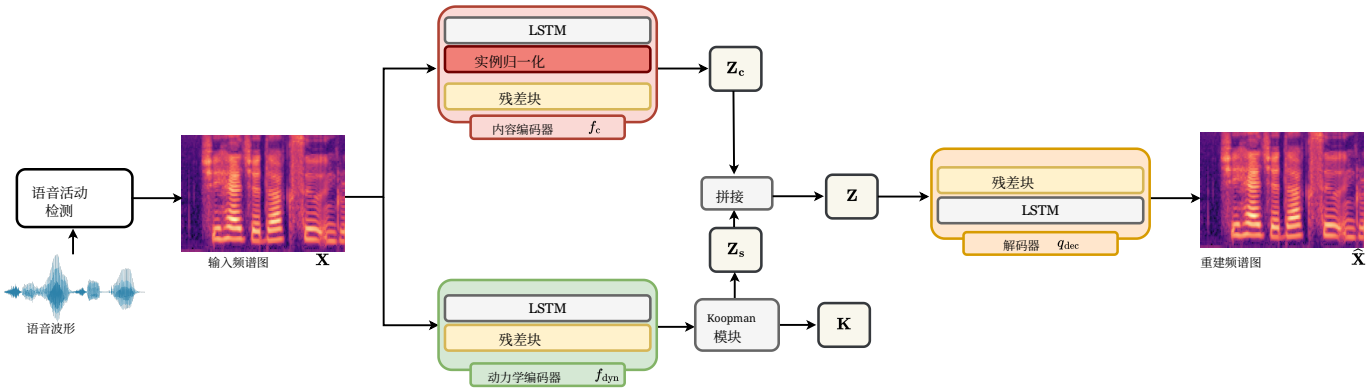

图 1. DKSD-AE 框架架构。语音话语经语音活动检测处理后，提取梅尔频谱图。随后，将输入梅尔频谱图 $\mathbf{X}$ 送入动力学编码器 f_{dyn}，以学习 Koopman 算子$\mathbf{K}$ 和说话人身份表示 $\mathbf{Z}_s$。与此同时，将 $\mathbf{X}$ 送入 f_c（内容编码器），通过实例归一化学习内容表示 $\mathbf{Z}_c$ 。最后，将 $\mathbf{Z}_s$ 和 $\mathbf{Z}_c$ 拼接后送入解码器 q_{dec}，生成重建梅尔频谱图 $\hat{\mathbf{X}}$。

在输入数据$\mathbf{X}$中。我们的模型通过内容编码器 f_c从输入 $\mathbf{X}$ 中学习内容表示$\mathbf{Z}_c \in \mathbb{R}^{B\times T\times k}$ ；该编码器受实例归一化影响，捕捉表征语音言语内容的时变属性。在以下各节中，为简化符号表示，呈现张量时我们将省略批次维度。图 1 展示了完整的 DKSD-AE 学习流程。

A. 用于解耦的 *Koopman* 算子学习

为了利用动力学编码器表示 $\mathbf{Z}_s$ 学习 Koopman 算子，我们采用了一种不同于文献中常用方法的方案。已有研究表明，使用 DMD 变体估计 $\mathbf{K}$ 会遇到计算可行性问题，例如过拟合；此外，任何噪声的存在都会影响从数据中提取有意义的动力学特征 [27], [28], [37]。为了使 $\mathbf{K}$ 的计算对训练早期阶段引入的噪声具有鲁棒性，我们对用于计算 $\mathbf{K}$ 的 Moore-Penrose 伪逆施加 ℓ_2 惩罚项。待最小化的目标函数为：

$$\mathbf{K} = \arg\min_{\mathbf{K}} \left\|\mathbf{Z}^{+} - \mathbf{Z}^{-}\mathbf{K}\right\|_F + \lambda \left\|\mathbf{K}\right\|_F^2, \quad \lambda \geq 0 \tag{6}$$

我们按如下方式求解公式（6）：

$$\mathbf{K} = ((\mathbf{Z}^{-})^T\mathbf{Z}^{-} + \lambda\mathbf{I})^{-1}(\mathbf{Z}^{-})^T\mathbf{Z}^{+} \tag{7}$$

此外，与常见的估计$\mathbf{K}$的方法不同，这些方法会将可观测函数$\mathbf{Z}$ 拆分为两个连续且重叠的窗口 [29]，[30]，，我们引入了一个多步预测 Koopman 近似模块，用于捕捉长期语音动力学。所提出的 Koopman 模块将$\mathbf{Z}_s$ 拆分为 M 个重叠窗口，窗口长度为 $\Delta t = 1$。我们认为，如果估计得到的$\mathbf{K}$ 能够准确预测未来的多个状态，那么我们的动力学编码器就学到了能够充分建模语音信号长期动态的可观测函数。为此，我们引入损失函数 $\mathcal{L}_{\text{pred}}$ ，该函数通过$\mathbf{K}$ 在 M步预测跨度内预测潜在可观测量空间中的状态来计算。令$\mathbf{Z}_s = f_{\text{dyn}}(\mathbf{X})$ $\in \mathbb{R}^{T\times k}$表示完整的动力学表示，其中 k 为潜在维度。为为了在长度为 M的预测跨度内进行多步预测，我们首先提取序列的一个时间前缀，

$$\mathbf{Z}_s^{\text{pre}} = \mathbf{Z}_s[0 : T - M] \in \mathbb{R}^{(T-M)\times k}, \tag{8}$$

$$T \in \mathbb{Z}_{>2}, \qquad M \in \{1, \ldots, T-2\}, \tag{9}$$

保留$\mathbf{Z}_s$前 $T - M$ 个时间步的副本。然后，通过构造两个时间平移张量来估计 Koopman 算子：

$$\mathbf{Z}_s^{-} = \mathbf{Z}_s[0 : T - M - 1], \quad \mathbf{Z}_s^{+} = \mathbf{Z}_s[1 : T - M], \tag{10}$$

这些张量用于按照公式（7）求解正则化伪逆。沿时间维度对$\mathbf{Z}_s$ 进行切片，并随后提取两个时间平移视图的过程如图2所示。接下来，使用$\mathbf{K}$ 预测未来 M 步的潜在可观测向量。对于每个预测时域 $m \in \{1, \ldots, M\}$，我们将预测的潜在动力学表示定义为：

$$\widehat{\mathbf{Z}}_s^{(m)} = \mathbf{Z}_s^{-}\mathbf{K}^m \in \mathbb{R}^{(T-M)\times k}, \tag{11}$$

其中，$\mathbf{K}^m$ 表示$\mathbf{K}$ 的 m 次幂。相应的真实目标张量为：

$$\mathbf{Z}_s^{(m)} = \mathbf{Z}_s[m : T - M + m - 1] \tag{12}$$

通过最小化预测残差的均方误差，可以促进 Koopman 算子$\mathbf{K}$ 的学习：

$$\mathcal{L}_{\text{pred}} = \frac{1}{2nM}\sum_{i}^{n}\sum_{m}^{M}\left\|\widehat{\mathbf{Z}}_{s,i}^{(m)} - \mathbf{Z}_{s,i}^{(m)}\right\|_2^2, \tag{13}$$

最小化 $\mathcal{L}_{\text{pred}}$ 可促使动力学编码器 f_{dyn} 学习 Koopman可观测表示，使潜在动力学在反复应用$\mathbf{K}$时呈线性演化。

作为对 $\mathcal{L}_{\text{pred}}$ 的补充，我们对 Koopman算子施加一个附加惩罚项，旨在约束其特征值谱，使其接近单位圆，并重点关注实单位特征值附近的区域，从而促进缓变动力学的建模。通过分析算子的特征值谱来识别以不同时间速率运行的属性，这种方法此前已有研究

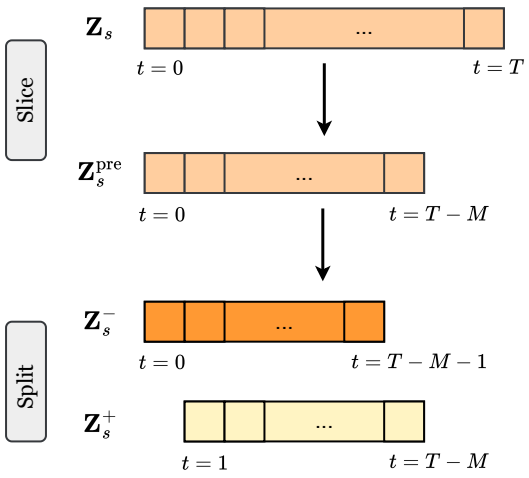

图2. 估计多步预测损失 $\mathcal{L}_{\text{pred}}$ 的处理步骤。从上到下，首先从完整的动力学表示 $\mathbf{Z}_s$中提取前缀 $\mathbf{Z}_s^{\text{pre}}$。然后将 $\mathbf{Z}_s^{\text{pre}}$拆分为两个时间上连续且错开的视图，用于求解公式（7）中所示的 Koopman算子回归。

用于背景建模 [38] 和多因素解缠 [34]。具体而言，众所周知，信号中随时间缓慢变化的特征对应于 Koopman算子的离散时间特征值 λ ，且这些特征值满足 $|\lambda| \approx 1$[38]。

我们重新审视相同的 [38] 概念，以约束 Koopman算子对随时间缓慢变化或保持静态的语音属性进行编码。为了鼓励学习到的 Koopman算子具有稳定的动力学，我们采用 SKD方法 [34]中引入的特征值惩罚。在训练过程中，估计出 $\mathbf{K}$ 后，我们计算其特征值分解，并使用特征值λ_i 计算特征值损失 $\mathcal{L}_{\text{eigen}}$ ，如下所示：

$$\mathcal{L}_{\text{eigen}} = \frac{1}{k}\sum_{i}^{k} |\lambda_i - (1+0j)|^2. \tag{14}$$

在 SKD 的公式中，该惩罚是施加于 $\mathbf{K}$ 的两种解耦约束之一；$\mathbf{K}$ 是从整体系统动力学中学习得到的算子。而在我们的模型中，它是施加于 $\mathbf{K}$ 的特征值谱上的唯一约束，该谱是在准静态语音动力学上估计得到的。这一设计选择反映了这样一个事实：f_{dyn} 仅编码准静态语音特征，因此在实践中，单一的特征值约束便已有效。特征值损失 $\mathcal{L}_{\text{eigen}}$ 旨在惩罚 Koopman 特征值偏离单位圆的情况，从而使其能够捕捉由说话人特征主导的静态和缓慢变化的语音特征。

B. 动力学编码器

所提出的动力学编码器 f_{dyn} 包含一叠LSTM [39] 块，其后是跳跃连接块 [40]，用于降低所生成动力学表示的维度。我们使用跳跃连接块，因为已有研究表明它们能够提升说话人验证系统 [41]中说话人表征的性能。残差或跳跃连接块允许在不降低其性能的情况下，通过估计残差函数，用额外层扩展网络：

$$f_{\text{skip}} = \mathbf{Z}_{\text{hid}} + h(\mathbf{Z}_{\text{hid}}), \tag{15}$$

其中， h(。)是任意激活函数，$\mathbf{Z}_{\text{hid}}$ 是前一层的输出。在 DKSD-AE 中， h(。)是双曲正切函数。因此，随着每增加一个模块，我们都会迭代学习更细粒度的表示，并避免训练过程中的计算不稳定问题。DKSD-AE 的动力学编码器会计算潜在可观测量，随后将其用作多步预测 Koopman 模块和相应 Koopman 损失函数 $\mathcal{L}_{\text{pred}}$ 和$\mathcal{L}_{\text{eigen}}$ 的输入。

C. 内容编码器

在初步实验中，我们探究了 SKD 框架 [34] 中的单算子Koopman表述能否实现稳定解耦。我们重新实现了 SKD 中描述的架构，并采用与 DKSD-AE 相同的预处理流程、训练协议和评估指标对其进行了评估。尽管 SKD 对图像序列有效，但在基于语音的实验设置下，我们未能复现其等错误率结果。这表明，约束单个 Koopman 算子的特征值谱可能无法充分捕捉高维语音信号中存在的多尺度结构。我们的观察与近期研究一致，这些研究指出，对多尺度非线性动力学建模可能需要表达能力更强或结构化程度更高的算子表述 [32], [35]。上述发现促使我们在模型设计中引入额外的归纳结构。

为了促进较慢和较快时间尺度的分离，我们引入了一个带有实例归一化层的额外编码器。根据经验，我们观察到，实例归一化有助于解耦语音话语中的潜在动力学，促使内容编码器捕获快速变化的语音属性，同时允许 f_{dyn}提取细粒度的说话人表征。内容编码器 f_{c} 由一组 LSTM 层 [39] 与实例归一化 [17],相结合，后接跳跃连接模块构成。自从在 [17] 中引入实例归一化以来，已有多种不同的实现方式被提出，它们会沿时间轴或频率轴对表示进行归一化 [42]。我们采用实例归一化模块，计算输入频谱图在频率维度上的均值和标准差，以消除与通道和说话人属性高度相关的全局逐语句统计量，具体如下：

$$\boldsymbol{\mu}_F = \frac{1}{F}\sum_{i=1}^{F}\mathbf{Z}_{f,:}, \quad \boldsymbol{\sigma}_F = \sqrt{\frac{1}{F}\sum_{f=1}^{F}(\mathbf{Z}_{f,:} - \boldsymbol{\mu}_F)^2 + \epsilon}, \tag{16}$$

其中 $\mathbf{Z}_{f,:}$ 表示频率箱 f处的所有时间索引。然后，每个实例归一化层块都会输出以下嵌入：

$$\mathbf{Z}_{\text{IN}} = \frac{\mathbf{Z} - \boldsymbol{\mu}_F}{\boldsymbol{\sigma}_F} \tag{17}$$

与 Chou 等人的实例归一化块不同， [17]我们的方案会先使用 LSTM 块处理频谱图，如图 1 所示。实例归一化在由两个编码器组成的 DKSD-AE 中有双重作用。它通过去除信道与说话人属性，使内容编码器 f_{c} 能够捕获语音内容信息。同时，它迫使动态编码器 f_{dyn}学习被丢弃的说话人属性，从而使解码器q_{dec} 能够充分重建输入频谱图 $\mathbf{X}$。

表 I DKSD-AE 架构的层数与隐藏节点维度。

组件	LSTM块	残差块
f_{c}	256,128,128,64	64,64
f_{dyn}	256,128	128,128,64,64,64,64,64
g_{dec}	128	64,64,128

D. 解码器

我们提出的架构中的解码器网络 q_{dec} 通过处理$\mathbf{X}$的输入频谱图在$\mathbf{Z}_c$ 和$\mathbf{Z}_s$ 沿潜在维度的拼接来重构输入频谱图:

$$\mathbf{Z} = \mathrm{concat}_{\mathrm{feat}}(\mathbf{Z}_s, \mathbf{Z}_c)\,, \tag{18}$$

该网络由一系列残差块构成，用于扩展$\mathbf{Z}$的特征维度，后接 LSTM 层。最后，一个线性层将 LSTM 隐藏状态投影到输入数据的频率箱维度。训练通过最小化重构频谱图的均方误差来实现，具体如下:

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{N}\sum_{i=1}^{N}\left\|\widehat{\mathbf{X}}_i - \mathbf{X}_i\right\|_2^2. \tag{19}$$

网络的总损失如下所示:

$$\mathcal{L}_{\mathrm{total}} = w_{\mathrm{rec}}\mathcal{L}_{\mathrm{rec}} + w_{\mathrm{pred}}\mathcal{L}_{\mathrm{pred}} + w_{\mathrm{eigen}}\mathcal{L}_{\mathrm{eigen}}, \tag{20}$$

其中， w_* 是可调超参数，用于加权总损失，以确保各项损失的数值尺度一致。

E. 用掩码增强捕捉说话人内变化

说话人验证的解纠缠既需要分离内容与身份属性，也需要充分表示说话人内变异性。动态编码器 f_{dyn} 旨在捕捉细粒度说话人身份线索，使同一说话人的表征在潜在空间中保持紧凑，同时与不同说话人的表征相互分离。为进一步实现对说话人内变化的编码，我们对输入频谱图采用 SpecAugment 时频掩码增强 [43]。SpecAugment 会沿语音频谱图的时间和频率维度，对相邻区域随机施加掩码，在训练过程中生成增强视图。这些增强视图使模型接触到能够保留说话人身份的扰动，从而促使编码器关注一致的说话人特有特征。

此外，由于增强会增加频谱图重构的难度，我们的模型首先在固定数量的训练轮次内进行预训练，且仅优化重构损失 $\mathcal{L}_{\mathrm{rec}}$ 。预训练的详细信息将在下一节中介绍。

V. 实验设置

A. 训练

训练期间，语音频谱图批次 $\mathbf{X}$ 会被输入两个编码器分支，以生成动力学表示$\mathbf{Z}_s$ 和语音内容表示 $\mathbf{Z}_c$。我们的模型经过提前停止训练，共训练 500 个训练轮次；前 30 个训练轮次仅优化重构损失 $\mathcal{L}_{\mathrm{rec}}$进行预训练。预训练后，模型针对完整损失 $\mathcal{L}_{\mathrm{total}}$进行优化。在完整训练期间，$\mathbf{Z}_s$ 与 Koopman模块结合，用于计算 Koopman算子估计 $\mathbf{K}$ ，具体如第IV-A节所述。随后，将$\mathbf{Z}_s$和$\mathbf{Z}_c$ 拼接后输入解码器 q_{dec} ，以生成重构结果 $\widehat{\ }\mathbf{X}$。我们使用 SpecAugment 变换 [43] 对数据进行增强，每个话语以 0.5 的概率通过增强函数进行修改。通过实验确定的各项损失权重分别为 $w_{\mathrm{rec}} = 1$、 $w_{\mathrm{pred}} = 0.1$和$w_{\mathrm{eigen}} = 5$。

我们提出方法的架构细节见表I。由f_{dyn} 得到的动力学表示形状为$\mathbf{Z}_s \in \mathbb{R}^{T\times B\times k}$ ，且 $k = 64$。估计的$\mathbf{K}$ 形状为 $k \times k$。关于由LSTM-IN块构成的内容编码器 f_c ，我们注意到每个LSTM的输出在送入下一层之前都会立即经过一个实例归一化块。此外，在解码器q_{dec}中，残差块位于LSTM块之前。

我们的模型使用AdamW优化器 [44] ，学习率为 10^{-4} ，权重衰减为 $\alpha = 0.4$。我们对LSTM以及残差块的线性层均使用双曲正切激活函数。最后，我们的Koopman模块将输入动态嵌入$\mathbf{Z}_s$ 划分为 $M = 5$个重叠的滑动窗口，并带有 $\Delta t = 1$。Koopman算子由窗口 m_1 和 m_2 计算得到，并在向前预测 M 步时对 $\mathcal{L}_{\mathrm{pred}}$ 进行评估。

B. 数据

我们在 VCTK [45]和 TIMIT [46] 数据集上评估了所提出的方法。VCTK 包含 110 位说话人，每位说话人约有 400 条话语。TIMIT 总计包含 630 位说话人，每位说话人有 10 条话语。为便于与现有 VCTK 基准进行一对一比较，我们将数据按 80% /10%/ 10% 划分为训练集、验证集和测试集，并将音频降采样至 16 kHz。我们的划分按说话人性别分层。对于 TIMIT，我们遵循官方划分，即 462 位说话人用于训练，24 位用于测试；同时也报告了包含 167 位说话人的完整测试集上的结果。两个数据集都经过基于自定义 WebRTC 的 [47] 语音活动检测（VAD）流程预处理，以从语音中提取纯语音片段。我们提取了 80 维对数梅尔频谱图，窗口长度为 50 ms，跳跃长度为 12.5 ms，并将其作为输入$\mathbf{X}$ 提供给模型。

表 II 所提出方法在说话人验证任务中相较于 VCTK 基线模型的最佳运行结果。无需文本标注或额外模态，DKSD-AE 在说话人验证任务中的表现优于其他竞争方法。各等错误率类别中的最佳表现以粗体标出。

方法	说话人等错误率 ↓	内容等错误率 ↑	模态类型	参数
DKSD-AE（我们的方法）	**2。77** %	44。0 %	梅尔频谱图	3.5M
SpeechTripleNet [20]	7.01 %	**45.5** %	梅尔频谱图、音高和能量轮廓	19.3M
VAE-TP [24]	2.90 %	44.1 %	梅尔频谱图和文本标注	399M
UTTS [5]	3.90 %	43.2 %	WavLM [4] 特征和文本标注	>94M

C. 指标

我们使用等错误率（EER）评估说话人验证性能。等错误率定义为误接受率（FAR）等于误拒绝率（FRR）时的工作点。假设$\mathbf{Z}_s$编码说话人特定信息，而$\mathbf{Z}_c$编码语言内容，那么如果使用$\mathbf{Z}_s$进行说话人验证时 EER 始终较低，而使用$\mathbf{Z}_c$时 EER 接近随机水平或显著变差，则表明解纠缠有效，这反映出内容表示中缺少说话人信息。计算 FAR 和 FRR 时，首先从话语中提取$\mathbf{Z}_s$和$\mathbf{Z}_c$。随后，我们计算同一说话人配对和不同说话人配对的余弦相似度分数。然后，在相似度分数上遍历决策阈值 $\epsilon \in [-1, 1]$。如果某次试验的相似度 $\geq \epsilon$，则将其分类为“接受”，否则分类为“拒绝”。对于每个 ϵ，FAR 和 FRR 按如下方式计算：

$$\mathrm{FAR}(\epsilon) = \frac{\#\text{imposter trials accepted at } \epsilon}{\#\text{total imposter trials}}, \tag{21}$$

$$\mathrm{FRR}(\epsilon) = \frac{\#\text{genuine trials rejected at } \epsilon}{\#\text{total genuine trials}}, \tag{22}$$

VI. 结果：说话人验证

A. 兼具鲁棒性与参数效率的说话人验证

我们使用等错误率作为评估指标，将 DKSD-AE 与最先进的说话人验证解耦模型进行比较。在各个数据集上，DKSD-AE 以显著更少的参数规模实现了具有竞争力或更优的说话人验证性能。表 II 和表 III 分别将我们表现最佳的模型与 VCTK 和 TIMIT 数据集上的基线方法进行比较，而表 IV 则展示了在 VCTK、TIMIT 官方数据集和 TIMIT 完整版上使用五组随机种子得到的结果，其中包括均值和标准差，以评估鲁棒性。

在 VCTK 上，我们针对基于不同归纳偏置进行条件化的解纠缠自编码器进行评估。具体而言，

表 III 我们的模型性能不逊于或优于最先进的方法，同时相比变分自编码器变体，在参数规模方面取得了显著改进。

方法	说话人等错误率 ↓	内容等错误率 ↑	参数
DKSD-AE（我们的）	3.90 %	**45。8** %	3.5M
DSVAE [13]	4.78 %	17.84 %	21M
C-DSVAE [23]	4.03 %	31.81 %	11M
SKD [34]	4.46 %	26.78 %	2M
D-DSVAE [22]	**3.25** %	38.83 %	*ND*

SpeechTripleNet [20]将梅尔频谱图与音高和能量特征相结合，VAE-TP [24] 利用文本标注学习文本条件先验以引导内容表示，UTTS [5] 则除文本监督外还依赖从大型预训练 WavLM 模型 [4] 中提取的自监督声学特征。对于 TIMIT，我们给出了与原始 DSVAE 及其扩展模型（即 C-DSVAE、D-DSVAE 和 SKD）的对比结果，这些模型在第二节中有更详尽的介绍。

表 II 和表 III 显示，在 VCTK 数据集上，DKSD-AE 的说话人等错误率低于所有基线方法；在 TIMIT 数据集上，除一种方法外，也低于所有其他方法。此外，这一性能是在可训练参数大幅减少、且不依赖梅尔频谱图以外的额外模态的情况下实现的，凸显了我们在高效说话人验证方面的设计所产生的影响。VCTK 和 TIMIT 上的趋势一致，表明所提出的归纳偏置能够泛化到录音条件不同的数据集。我们的模型在内容等错误率方面也优于除一种方法外的所有基线方法，进一步验证了解纠缠对于说话人验证而言并无信息价值这一点。

此外，表 IV 报告了 VCTK、TIMIT 和规模大得多的 TIMIT-full 测试集上的说话人等错误率和内容等错误率，并给出了不同随机种子下的均值和标准差。有趣的是，**尽管评估规模从 TIMIT 官方测试集扩大到 TIMIT-Full测试集，增幅近七倍，说话人等错误率仅 ≈恶化了 1%**，这表明所提出的表示具有良好的泛化能力，并且随着说话人数增加仍保持稳定。进一步报告五个不同随机种子下的平均等错误率表现及相应标准差，也证明了所提方法的稳定性：各数据集上观察到的标准差始终较低，表明该方法对随机初始化的敏感性有限。

除等错误率结果外，我们还通过对学习到的解纠缠表示进行可视化，验证所提出的解纠缠方法的有效性。从测试说话人的话语中提取说话人表示和内容表示，并使用主成分分析和 t-SNE 将其投影到二维空间，以供比较 [48]。图 3 显示，说话人表示$\mathbf{Z}_s$为每个说话人类别形成了定义清晰的簇，而内容嵌入$\mathbf{Z}_c$则没有呈现说话人特定的结构，而是散布在二维平面上，表明解纠缠成功。

我们的方法仅依赖语音数据，因此 DKSD- AE 更加灵活，只需进行少量调整，便可应用于不同的语音数据集。相比之下，在无法

表 IV DKSD-AE 在不同随机初始化下，在 VCTK、TIMIT 和 TIMIT-TEST 上均表现出稳定的等错误率性能。

数据集	说话人等错误率 ↓	内容等错误率 ↑
VCTK	3.44 ± 0.56 %	44.70 ± 1.50 %
TIMIT 官方数据集	4.68 ± 0.69 %	46.04 ± 0.59 %
TIMIT 完整集	5.64 ± 0.27 %	45.94 ± 0.62 %

获取文本数据时，需要同时使用音频和文本模态的方法只能应用于带有对齐转录文本的数据集，否则可能无法直接使用。UTTS [5], 的情况便体现了这一点：由于 VCTK 语料库中没有说话人“p315”的转录文本，作者在实验中将其排除。我们的方法无需文本标签，计算开销也比对比方法更低，因此可以部署在资源受限设备上。

B. 正则化算子理论表示增强语音解耦

我们通过比较用于解纠缠的 Koopman 模块的不同实现，展示了利用我们提出的 $\mathcal{L}_{\text{total}}$ 学习算子理论表示所带来的解纠缠性能提升。具体而言，我们将所提出的模型与不含特征值惩罚$\mathcal{L}_{\text{eig}}$的版本（该版本仅学习一个具有 $\mathcal{L}_{\text{pred}}$ 的 Koopman 算子），以及不进行 Koopman 学习的版本进行比较。表 V 显示，采用完整目标函数 $\mathcal{L}_{\text{total}}$ 进行训练可获得最佳的说话人和内容等错误率表现。结果表明，仅靠实例归一化就能使内容编码器 f_c捕获语音内容，从而迫使 f_{dyn} 编码高层说话人属性。然而，**通过条件化 f_{dyn} 来学习具有多步时间一致性的 Koopman 可观测函数，可以学习更细粒度的说话人特征**，并且与仅使用实例归一化的变体相比，可减少慢特征泄漏到内容表示$\mathbf{Z}_c$ 中。

语音包含以不同时间尺度演变的特征。我们通过实例归一化对快速变化的属性进行建模，并

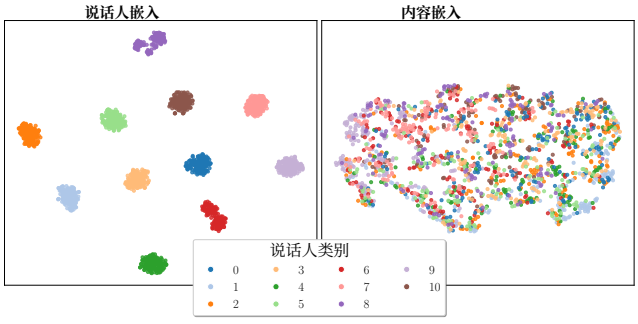

图 3. 使用主成分分析和 t-SNE [48] 降维后，说话人表示 $\mathbf{Z}_s$ （左）与内容表示 $\mathbf{Z}_c$ （右）的可视化对比。相同说话人的表示 $\mathbf{Z}_s$ （左）在二维空间中形成紧凑且彼此分离的簇，而内容表示 $\mathbf{Z}_c$ 则分散分布，不呈现说话人特定的分组（右）。不同颜色对应不同的说话人类别。

表 V 在不同数据集和不同随机种子下，与仅重构基线以及仅使用 $\mathcal{L}_{\text{PRED}}$ 训练的 Koopman 自编码器相比，完整目标函数带来了更优的说话人验证性能和更强的解纠缠。

数据集	Koopman 学习	说话人等错误率	内容等错误率
VCTK	$\mathcal{L}_{\text{total}}$	$\mathbf{3.66} \pm 0.55$ %	$\mathbf{44.36} \pm 0.93$%
	$\mathcal{L}_{\text{pred}}$	5.27 ± 0.49 %	35.04 ± 3.97%
	$\mathcal{L}_{\text{rec}}$	4.65 ± 0.25 %	39.04 ± 1.95%
TIMIT	$\mathcal{L}_{\text{total}}$	$\mathbf{4.68} \pm 0.69$ %	$\mathbf{46.04} \pm 0.59$ %
	$\mathcal{L}_{\text{pred}}$	7.15 ± 0.50 %	39.82 ± 1.98 %
	$\mathcal{L}_{\text{rec}}$	6.47 ± 1.18 %	43.39 ± 3.38 %

通过在 f_{dyn} 中进行多步 Koopman 预测，对缓慢变化的说话人动态进行建模，从而实现分层解耦。正如近期文献 [32], [35] 所指出的，期望单个线性算子能够高效地建模高维、多尺度语音特征的传播，存在显著局限。不同于先前使用 Koopman 嵌入对语音序列进行解纠缠的工作 [34], ，我们不使用单个算子对系统进行建模。

C. 多步 *Koopman* 损失使语音长期动态捕获成为可能

本节展示，所提出的多步预测损失 $\mathcal{L}_{\text{pred}}$ 使 Koopman 算子能够捕获语音中的长期依赖关系。我们通过一项消融研究验证了这一点：在 TIMIT 和 VCTK 上，与单步 $\mathcal{L}_{\text{pred}}$ 方法相比，多步训练能够降低说话人等错误率。消融结果见图 4。我们展示了采用相同配置、仅在 M的选择上有所不同的模型所生成的说话人嵌入$\mathbf{Z}_s$ 的等错误率。对于 VCTK 数据集，预测时域最多达到向前 $M = 15$ 步；对于 TIMIT，则达到 $M = 13$ 步时，多步 $\mathcal{L}_{\text{pred}}$ 模型的表现始终优于 Koopman 自编码器方法，其中 $M = 1$，在解纠缠与等错误率方面均如此。当使用 $M = 1$ 近似$\mathbf{K}$ 时，估计 Koopman 算子的可观测函数只需捕获向前一步的动态，即可确保单步$\mathcal{L}_{\text{pred}}$收敛；但这可能导致编码器在有限的预测时域上过拟合，而忽略过去或未来的动态。相比之下，尽管我们根据$\mathbf{Z}_s^{\text{pre}}$的前两个滑动窗口学习$\mathbf{K}$ ，如式（10）所示，但我们的 $\mathcal{L}_{\text{pred}}$ 是在向前 M 个状态处评估的，使 f_{dyn} 能够学习捕获长期动态的可观测函数。

$\mathbf{K}$ 的多步时间一致性是解纠缠的强大正则化项，无论底层采用何种动力学参数化。多步 $\mathcal{L}_{\text{pred}}$的优势在不同数据集上都表现一致，不过在 TIMIT 上不太明显，这可能是因为其说话人数量多于 VCTK。图 4 还显示了一个有效的 M范围，在此范围内，可利用 Koopman 可观测函数高效地线性化系统动力学，从而在下游表征学习任务中取得良好性能。当 M 超过所确定的范围时，说话人验证性能会下降，因为在高维潜在空间中进行多步预测将变得愈发困难。

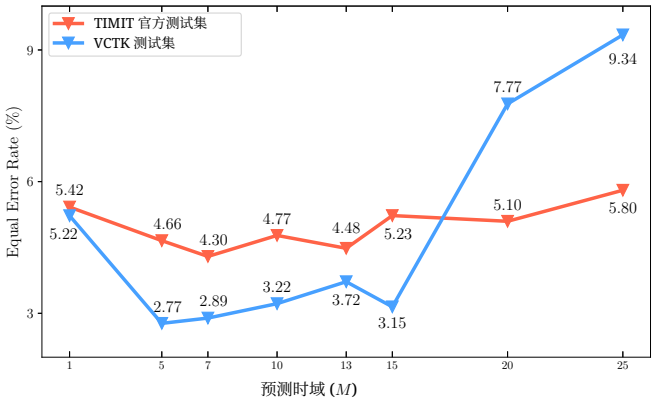

图 4. 我们的多步 Koopman 算子学习模块公式表明，在 VCTK 上，当预测时域为 $M \in [5, 13]$ 时，说话人等错误率有所改善；在 TIMIT 上也有改善，但不太明显。所确定的有效 M 范围对应于预测时域长度为初始语音话语时长的 5% 至 10%。

VII. 结论

本文提出了一种深度自编码器，用于将语音中的说话人身份与言语内容表示解纠缠，以实现说话人验证。我们提出了 DKSD-AE，将多步 Koopman 算子学习和实例归一化作为归纳偏置，以促进变化迅速的内容动态与缓慢演变的说话人相关属性分离。实验评估表明，该方法得到的说话人表示具有良好的验证判别性，且不受语言内容影响。与基线方法相比，该方法取得了具有竞争力或更优的说话人验证性能，同时内容等错误率较高；此外，其参数量更少，且不依赖文本监督。在评估规模扩大以及跨多个数据集的情况下，所提出的方法依然稳健，彰显了其泛化能力。总体而言，我们的研究结果表明，结构化的 Koopman 算子学习结合实例归一化，为解纠缠说话人属性与语音内容属性提供了一条高效且有原则的方法。

当前评估仅限于文本无关说话人验证，将其扩展到情感语音或声学退化条件仍是未来的工作。下一步自然是将所提出的基于 Koopman 的动态模块与 Transformer 式编码器相结合， [49],利用其强大的序列建模能力，扩展到更长且更多样化的话语，同时保留所提出的归纳偏置。

致谢

本研究得到英国研究与创新署纳米电子器件与系统机器智能博士培训中心 [EP/S024298/1] 以及工程与物理科学研究委员会（EPSRC）ActivATOR项目 [EP/W017466/1]的资助。作者感谢在完成本工作过程中使用了南安普顿大学的IRIDIS高性能计算设施及相关支持服务。

参考文献

[1] 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深度神经网络嵌入”，ICASSP，2018年，第5329–5333页。[2] B. Desplanques、J. Thienpondt和K. Demuynck，“ECAPA-TDNN：基于TDNN的说话人验证中的强化通道注意力、传播与聚合”，Interspeech，2020年，第3830–3834页。[3] W.-N. Hsu、B. Bolte、Y.-H. H. Tsai、K. Lakhotia、R. Salakhutdinov和A. Mohamed，“HuBERT：通过隐单元掩码预测进行自监督语音表示学习”，IEEE/ACM《音频、语音与语言处理汇刊》，第29卷，第3451–3460页，2021年。[4] 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《选题信号处理杂志》，第16卷，第6期，第1505–1518页，2022年10月。[5] J. Lian、C. Zhang、G. K. Anumanchipalli和D. Yu，“面向TTS的无监督TTS声学建模：条件解耦序列变分自编码器”，IEEE/ACM《音频、语音与语言处理汇刊》，第31卷，第2548–2557页，2023年。[6] Z. Li、M.-W. Mak、J.-T. Chien、M. Pilanci、Z. Jin和H. Meng，“利用潜在扩散解纠缠语音表征学习以用于说话人验证”，IEEE/ACM《音频、语音与语言处理汇刊》，第33卷，第3896–3907页，2025年。[7] P. Rose，《法庭说话人识别》。伦敦：CRC出版社，2002年7月。[8] X. Wang、H. Chen、S. Tang、Z. Wu和W. Zhu，“解耦表示学习”，IEEE《模式分析与机器智能汇刊》，第46卷，第12期，第9677–9696页，2024年12月。[9] D. P. Kingma和M. Welling，“自编码变分贝叶斯”，国际学习表征会议论文集，2014年。[10] F. Locatello、S. Bauer、M. Lucic、G. Raetsch、S. Gelly、B. Sch¨olkopf和O. Bachem，“挑战解耦表示无监督学习中的常见假设”，国际机器学习会议论文集，2019年，第4114–4124页。[11] A. van den Oord、O. Vinyals和K. Kavukcuoglu，“神经离散表示学习”，国际神经信息处理系统会议论文集，2017年，第6309–6318页。[12] W.-N. Hsu、Y. Zhang和J. Glass，“从序列数据中无监督学习解耦且可解释的表示”，国际神经信息处理系统会议论文集，2017年，第1876–1887页。[13] L. Yingzhen和S. Mandt，“解耦序列自编码器”，国际机器学习会议论文集，第80卷，2018年，第5670–5679页。[14] B. O. Koopman，“哈密顿系统与希尔伯特空间中的变换”，《美国国家科学院院刊》，第17卷，第5期，第315–318页，1931年5月。[15] C. W. Rowley、I. Mezic、S. Bagheri、P. Schlatter和D. S. Henningson，“非线性流的谱分析”，《流体力学杂志》，第641卷，第115–127页，2009年。[16] D. Ulyanov、A. Vedaldi和V. Lempitsky，“实例归一化：快速风格化所缺失的要素”，2017年，arXiv:1607.08022。[17] J. C. Chou和H. Y. Lee，“通过实例归一化分离说话人与内容表示的单次语音转换”，Interspeech，2019年，第664–668页。[18] Y. H. Chen、D. Y. Wu、T. H. Wu和H. Y. Lee，“Again-VC：一种使用激活引导与自适应实例归一化的单次语音转换”，ICASSP，2021年，第5954–5958页。[19] W.-C. Huang、H. Luo、H.-T. Hwang、C.-C. Lo、Y.-H. Peng、Y. Tsao和H.-M. Wang，“基于变分自编码器的语音转换中利用跨域特征与对抗学习的无监督表示解纠缠”，IEEE《计算智能新兴主题汇刊》，第4卷，第4期，第468–479页，2020年8月。[20] H. Lu、X. Wu、Z. Wu和H. Meng，“SpeechTripleNet：面向内容、音色与韵律的端到端解耦语音表征学习”，ACM国际多媒体会议论文集，2023年，第2829–2837页。[21] N. Gengembre、O. Le Blouch和C. Gendrot，“通过语音转换解纠缠韵律与音色嵌入”，Interspeech，2024年，第2765–2769页。[22] J. Lian、C. Zhang和D. Yu，“面向零样本语音转换的鲁棒解耦变分语音表征学习”，ICASSP，2022年，第6572–6576页。[23] J. Bai、W. Wang和C. Gomes，“对比解耦序列变分自编码器”，《神经信息处理系统进展》，2021年，第10105–10118页。

[24] H. Lu、X. Wu、H. Guo、S. Liu、Z. Wu和H. Meng，“利用解耦语音表征统一单次语音转换与克隆”，ICASSP，2024年，第11141–11145页。[25] P. J. Schmid，“数值与实验数据的动态模态分解”，《流体力学杂志》，第656卷，第5–28页，2010年。[26] M. O. Williams、I. G. Kevrekidis和C. W. Rowley，“Koopman算子的数据驱动近似：扩展动态模态分解”，《非线性科学杂志》，第25卷，1346，12 2015年 6，1307 no. pp. – 。[27] S. T. M. Dawson、M. S. Hemati、M. O. Williams和C. W. Rowley，“刻画并校正动态模态分解中传感器噪声的影响”，《流体实验》，第57卷，第3期，第42页，2016年2月。[28] S. L. Brunton、M. Budiˇsi´c、E. Kaiser和J. N. Kutz，“面向动力系统的现代Koopman理论”，《SIAM评论》，第64卷，第2期，第229–340页，2022年。[29] N. Takeishi、Y. Kawahara和T. Yairi，“学习用于动态模态分解的Koopman不变子空间”，《神经信息处理系统进展》，2017年，第1131–1141页。[30] B. Lusch、J. N. Kutz和S. L. Brunton，“面向非线性动力学通用线性嵌入的深度学习”，《自然·通讯》，第9卷，第1期，第1–10页，2018年11月。[31] O. Azencot、N. B. Erichson、V. Lin和M. Mahoney，“使用一致性Koopman自编码器预测序列数据”，国际机器学习会议论文集，2020年，第475–485页。[32] N. Geneva和N. Zabaras，“用于物理系统建模的Transformer”，《神经网络》，第146卷，第272–289页，2022年2月。[33] Y. Liu、C. Li、J. Wang和M. Long，“Koopa：利用Koopman预测器学习非平稳时间序列动力学”，《神经信息处理系统进展》，2023年，第12271–12290页。[34] N. Berman、I. Naiman和O. Azencot，“通过结构化Koopman自编码器实现多因素序列解纠缠”，国际学习表征会议论文集，2023年，第1–25页。[35] H. Bai、W. Ding和D. Zou，“分层Koopman扩散：借助可解释扩散轨迹的快速生成”，《神经信息处理系统进展》，2025年10月。[36] R. Wang、Y. Dong、S. Arik和R. Yu，“面向具有时间分布偏移的时间序列的Koopman神经预测器”，国际学习表征会议论文集，2023年。[37] S. E. Otto和C. W. Rowley，“用于学习动力学的线性循环自编码器网络”，SIAM《应用动力系统杂志》，第18卷，第1期，第558–593页，2019年3月。[38] N. B. Erichson、S. L. Brunton和J. N. Kutz，“用于背景建模的压缩动态模态分解”，《实时图像处理杂志》，第16卷，第5期，第1479–1492页，2019年10月。[39] 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11月。[40] K. He、X. Zhang、S. Ren和J. Sun，“用于图像识别的深度残差学习”，IEEE计算机学会计算机视觉与模式识别会议论文集，2016年，第770–778页。[41] C. Li、X. Ma、B. Jiang、X. Li、X. Zhang、X. Liu、Y. Cao、A. Kannan和Z. Zhu，“Deep Speaker：端到端神经说话人嵌入系统”，2017年，arXiv:1705.02304。[42] T. Lertpetchpun和E. Chuangsuwanich，“用于说话人验证的基于实例的时间归一化”，Interspeech，2023年，第3172–3176页。[43] D. S. Park、W. Chan、Y. Zhang、C.-C. Chiu、B. Zoph、E. D. Cubuk和Q. V. Le，“SpecAugment：一种用于自动语音识别的简单数据增强方法”，Interspeech，2019年，第2613–2617页。[44] I. Loshchilov和F. Hutter，“解耦权重衰减正则化”，国际学习表征会议论文集，2017年。[45] J. Yamagishi、C. Veaux和K. MacDonald，“CSTR VCTK语料库：面向CSTR语音克隆工具包的英语多说话人语料库（0.92版）”，2019年11月。[46] Garofolo、John S.、Lamel、Lori F.、Fisher、William M.、Pallett、David S.、Dahlgren、Nancy L.、Zue、Victor和Fiscus、Jonathan G.，“TIMIT声学语音连续语料库”，1993年。[47] 谷歌公司，“WebRTC语音活动检测”，https://webrtc.org/，访问日期：2026年1月。[48] L. v. d. Maaten和G. Hinton，“使用t-SNE可视化数据”，《机器学习研究杂志》，第9卷，第86期，第2579–2605页，2008年。[49] S. Latif、S. A. M. Zaidi、H. Cuay´ahuitl、F. Shamshad、M. Shoukat、M. Usama和J. Qadir，“语音处理中的Transformer：克服挑战并开辟未来”，《计算机科学评论》，第58卷，文章编号100768，2025年11月。