面向跨域语音深度伪造检测的冻结的自监督学习编码器与后端特性分析

Benjamin Hurt*、Oscar O'Donnell†‡、Yefei Yang*‡、Anjani Sethi*、Priyanka Durai*
*加利福尼亚大学圣塔芭芭拉分校，美国
†哥伦比亚大学，美国纽约
‡这些作者对本研究的贡献相同。电子邮箱： {hurt、yefei yang、anjanisethi、priyankadurai}@ucsb.edu、ojo2013@columbia.edu

arXiv:2611.29865v1 [eess.AS] 23 Nov 2026

摘要——自监督学习（SSL）编码器是语音深度伪造检测中占主导地位的前端，而当前普遍采用的方案是在小规模欺骗语音语料库上对完整编码器进行微调，以域内准确率最大化为代价，换取已知的域偏移下性能下降。我们将近期一种融合 SSL 与常数Q倒谱特征的交叉注意力融合方案作为固定参考方案，冻结其中的编码器，并考察哪种编码器和哪种后端在跨领域场景下表现最佳。我们将两个冻结编码器——大规模多语言 XEUS 和 Wav2Vec2 XLSR-53——分别与两个后端，即多头因子化注意力（MHFA）池化和 AASIST，进行组合，并在 ASVspoof 2019 LA、ASVspoof 2021 LA 和 DF、In-the-Wild（ITW）以及 ASVspoof 5 上进行评估；所有系统均在同一流水线中使用三个随机种子进行训练。我们报告三项发现。第一，编码器与后端存在交互作用：MHFA 更适合性能更强的编码器，而 AASIST 能够从较弱的编码器中恢复更多性能，并使其表现更加稳定。第二，在该方案中，冻结编码器会产生显著且可重复的差异。使用 MHFA 时，在 LA21、DF21 和 ASV5 上，冻结 XEUS 相较于冻结 XLSR-53 可将错误率降低 58%–77%，在 ITW 上的改善幅度则较为有限。在匹配聚合消融实验中，这一排序仍然保持不变；此外，参数量匹配的 WavLM Large 诊断实验表明，参数量本身无法单独解释 XLSR-53 较弱的性能。第三，与使用三个随机种子的微调基线相比，冻结 XEUS 仅更新不到 0.4M 个参数，而基线更新 316M 个参数；冻结 XEUS 在 ASVspoof 5 上（就等错误率和最小检测代价函数而言）以及 LA21 上取得了小幅优势，但在 DF21 和 ITW 上落后，并且在四个域外数据集中的三个上表现出更低的随机种子间变化。

索引词—语音深度伪造检测、反欺骗、自监督学习、冻结编码器、跨域评估、训练效率

I. 引言

文本转语音和语音转换技术的进步，使得合成语音难以与真实录音区分开来，从而推动了大量关于语音深度伪造检测和反欺骗的研究。该领域已形成一种标准架构：大型自监督学习（SSL）语音编码器生成帧级表示，再由轻量级后端将其分类为真实语音或伪造语音[1], [2]。主流方案是在欺骗语音语料库（通常为 ASVspoof 2019 LA）上对自监督学习编码器进行端到端微调。微调能够取得出色的域内等错误率（EER），但也会使通用编码器适应单一语料库的狭窄统计特征；据广泛报道，以这种方式训练的检测器容易对已知攻击过拟合，并且难以迁移到未见过的攻击、编解码器和录音条件 [1], [3], [4]。In-the-Wild 基准使这一差距具体显现出来：在 ASVspoof 上表现良好的检测器在收集到的真实世界音频上，性能最多会下降一个数量级 [4]。

近期的一种融合框架——其发起者将其称为交叉注意力基线 [5]，——通过交叉注意力，将微调的 Wav2Vec2 XLSR-53 前端与手工设计的常数 Q 倒谱系数（CQCC）以及 AASIST后端相结合。其作者指出了两个未来研究方向：探索“不同的后端分类器”，以及向融合中添加韵律描述符 [5]。我们直接着手研究第一个方向，并以第二个方向作为研究动机。我们没有将韵律特征附加到现有编码器上，而是考察这样一个问题：当两者均被冻结时，一个其预训练本身已涵盖韵律多样化材料的编码器 XEUS [6]（在 4,057 种语言中包含超过一百万小时的数据，语料库中包含歌声，并采用了以声学鲁棒性为目标的去混响目标），是否比 XLSR-53 [7] 更适合作为固定表征。近期一项关于人工智能生成的歌声的研究发现，在语音嵌入方面，XEUS 的表现优于 Wav2Vec；其作者将这一结果归因于能够捕捉韵律和音色的预训练分布 [8]。

冻结编码器将问题从适应重新定义为表征。在编码器固定的情况下，系统之间的差异反映了：(i) 给定的预训练分布捕捉了什么，以及 (ii) 后端如何读取这些信息。已有研究表明，在某些设置下，未经微调的前端仍具有竞争力 [2]，然而，冻结的 SSL 检测器在域偏移下仍然难以应对，其准确率会因编码器和测试语料库而显著变化 [9]。这正是共享流程比较冻结编码器所能刻画的内容，也回应了这样一种认识：仅凭较低的错误率并不能确立可信度 [10]–[11]。我们选择冻结，而不是使用 LoRA 或适配器进行适应 [12]，因为任何适应都会使预训练与特定语料库的调优重新纠缠在一起。冻结编码器比较的是其预训练时形成的表征。冻结操作会从比较中移除特定语料库的适应，但并不能将预训练分布与目标函数或规模区分开来。第五节 E 使用容量匹配编码器探究规模因素。因此，我们的冻结系统并不以达到最先进的等错误率为目标；我们关注的是哪些表征能够跨数据集迁移，以及需要多少可训练参数，并据此限定我们的论断范围。

我们的贡献如下：

- 一种共享流水线刻画，适用于已发表的

© 2026年 IEEE。允许个人使用本材料。对于任何其他用途，包括在任何当前或未来媒介中为广告或宣传目的重印／再版本材料、创作新的合集作品、转售或重新分发至服务器或列表，或在其他作品中重复使用本作品的任何受版权保护部分，均须事先获得 IEEE 的许可。

交叉注意力融合方案 [5] 保持固定，并将两个冻结的自监督学习编码器（XEUS、XLSR-53）与两个后端（MHFA、AASIST）进行交叉组合，涵盖五个基准测试，涉及域内、近域、真实世界以及大规模众包条件。

- 编码器与后端并非各自独立发挥作用的证据：表现更强的后端取决于其读取的编码器（MHFA 更适合更强的编码器，而 AASIST 能从较弱的编码器中获得更多收益并提升其稳定性），因此后端的选择无法跨编码器迁移。

- 在这一固定方案中，冻结编码器选择会导致显著且可重复的跨数据集差异，且这种排序在匹配聚合消融实验中仍然保持。容量匹配的 WavLM Large 诊断实验进一步表明，XLSR-53 表现较弱不能仅用参数量来解释。我们明确指出预训练/评估来源重叠这一情况，并将其视为解读这些差异是否体现纯粹分布泛化能力的一项限制。

- 在流水线内与采用三个随机种子的全编码器微调基线进行比较：冻结的广覆盖编码器更新的参数不足 0.4M，而基线为 316M；该编码器在 LA21 和 ASV5 上表现更好，但在 DF21 和 ITW 上表现较差，并且在四个域外数据集中的三个数据集上观察到的种子间变化较小。

II. 相关工作

用于反欺骗的自监督学习前端。Wav2Vec2 XLSR-53 [7] 以及相关模型通常经过微调，构成了近期大多数检测器的基础；Tak等人 [1]提出的 wav2vec2-plus-AASIST 方案具有代表性，其动机源于传统反制措施泛化能力较差。Wang和 Yamagishi [2]报告称，未经微调的前端也具备竞争力，但为了获得最佳域内结果，仍倾向于采用微调，这促成了我们的冻结设计。XEUS [6] 是较新的大规模多语言编码器，其预训练着重于声学鲁棒性。据我们所知，此前唯一将其用于反欺骗的工作，是在一项前端比较中将其作为前端使用，但据报道并未改善结果，且未公布具体数值[13], ，因此针对其与 XLSR-53 的系统性冻结比较仍有待开展。近期一项针对二十种冻结的自监督学习前端的基准研究发现，大型多语言判别式编码器属于表现最强的模型，并采用了我们使用的冻结编码器、简易后端范式，但该研究未包含 XEUS [14]。

后端。 AASIST [15]将时频图注意力与注意力统计池化相结合，是交叉注意力基线 [5]中的后端。MHFA [16],最初用于说话人验证，它在编码器层上学习独立的键和值权重，并通过多个因子化头进行池化。该方法被提出作为一种用于预训练编码器的高效后端。我们在同一个融合架构中对这两种方法进行比较。

层聚合。 SSL 层编码不同的信息 [17], ，而在深度伪造检测中，最具判别力的欺骗线索集中于中下层 Transformer 层 [18]。这促使我们将层聚合视为

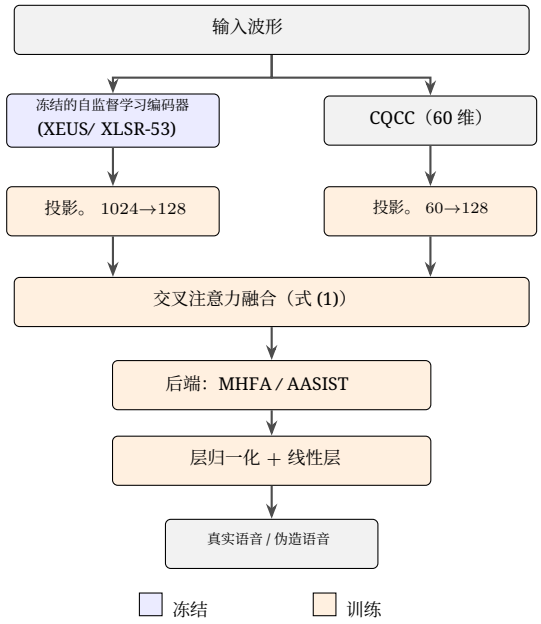


图 1. 冻结编码器交叉注意力融合。自监督学习编码器（蓝色）被冻结；只有投影层、融合模块、后端和分类器（橙色）参与训练。自监督学习流查询频谱流，残差连接则保留自监督学习摘要（公式 (1)）。我们将两个编码器与两个后端交叉组合，得到所研究的四种系统。

每个编码器配置的一部分，而非自由变量，并在 V-B 节中单独考察其影响。

参数高效检测。基于冻结基础模型构建紧凑型后端是一个活跃的研究方向。Nes2Net [19]指出，后端中的降维层会造成信息损失，同时增加参数开销，并提出了一种更轻量的嵌套式设计。Nes2Net 在固定编码器上对后端进行基准测试，而我们关注的是与之正交的另一个维度：应固定哪种冻结编码器，并通过析因设计和匹配聚合消融实验进行研究。

域泛化。另一条互补的研究路线通过显式特征对齐来应对域偏移：借助梯度反转获得语料不变特征 [20],、对自监督学习嵌入进行无监督统计适配 [21], ，以及记录真实世界条件下性能退化情况的多语言 ML-ITW基准 [22]。域外失效究竟反映了攻击本身确实更难，还是仅仅反映攻击类型不同，这一问题也已得到直接研究 [23]。这些事后补救方法带来的提升有限且参差不齐。我们转而探究，能否通过选择预训练分布，使鲁棒性成为冻结表示本身的属性，而非纠正性适配步骤的结果。

III. 方法

A. 交叉注意力融合架构

该架构如图1所示。对于输入波形，我们提取两条流。自监督学习流应用冻结编码器 $E \in \{\text{XEUS}, \text{XLSR-53}\}$ 以获取帧特征$\mathbf{f}_{\text{ssl}} \in \mathbb{R}^{T_{\text{ssl}} \times 1024}$ （两个编码器均为1024维）。频谱流计算60维CQCC特征 [24]（从每倍频程96个频点的常数Q变换中提取20个倒谱

系数，其中 Δ 和 ΔΔ），$\mathbf{f}_{\text{sf}} \in \mathbb{R}^{T_{\text{sf}} \times 60}$。线性投影层将两条流映射到共同维度 $D=128$，并通过线性插值将频谱流的长度匹配到自监督学习流的帧数 $T_{\text{ssl}}=T$，得到 $\tilde{\mathbf{f}}_{\text{ssl}}$，$\tilde{\mathbf{f}}_{\text{sf}} \in \mathbb{R}^{T \times D}$。

融合采用交叉注意力，以投影后的自监督学习流作为查询，以投影后的频谱流作为键和值，并通过残差连接保留自监督学习摘要：

$$\mathbf{H} = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\top}}{\sqrt{D}}\right)\mathbf{V} + \tilde{\mathbf{f}}_{\text{ssl}}, \tag{1}$$

其中 $\mathbf{Q}=\tilde{\mathbf{f}}_{\text{ssl}}W_Q$、$\mathbf{K}=\tilde{\mathbf{f}}_{\text{sf}}W_K$ 和 $\mathbf{V}=\tilde{\mathbf{f}}_{\text{sf}}W_V$ 分别与W_Q、W_K、$W_V \in \mathbb{R}^{D \times D}$结合，因此每一项的 $\mathbb{R}^{T \times D}$ 均保持一致，残差连接在维度上也是一致的。融合序列$\mathbf{H}$ 被传递至后端，生成话语嵌入，随后经过层归一化和线性层真实/欺骗分类器。

B. 后端

我们比较了两种作用于$\mathbf{H}$的后端。*MHFA* [16]构成键和值，使用 $H=8$ 因子化头对$K=64$压缩后的成分进行池化，并输出一个128维嵌入。*AASIST* [15] 应用其时频图注意力堆叠。与编码器相比，两者都较为轻量。

C. 冻结因子实验

我们将编码器 $\in$ {XEUS、XLSR-53} 分别与后端$\in$ {MHFA、AASIST}进行交叉组合，得到四个系统。在每个实验单元中，编码器均被冻结，仅训练投影层、交叉注意力、后端和分类器。对于哪些因素匹配、哪些因素不匹配，我们进行了明确区分。各实验单元匹配的部分包括：编码器的冻结状态、CQCC频谱分支、预处理、公式（1）中的融合机制以及评估协议；编码器和后端是两个实验因素。不匹配的部分则被视为每个编码器最佳实践配置的一部分，而不是受控变量：SSL 层聚合（XEUS 最后一层与 XLSR-53 最后四层的均值，依据逐层证据[17], [18]为每个编码器分别选择）、训练轮数以及 MHFA 实验单元的学习率调度方案。因此，我们将主要析因设计呈现为在合理配置下对各编码器的比较，而不是单变量消融。V-B 节单独考察层聚合因素，其余因素将在第 VI 节讨论。我们固定采用交叉注意力 CQCC 融合，因为实验证明它是基线的最佳变体[5]；消融频谱分支与编码器问题正交。

我们使用 XLSR-53 作为比较对象，因为它是参考方案 [5] 固定采用的前端。我们明确指出，这一选择带来的结果是：我们的编码器比较仅限于该方案内部。它反映的是在一个固定的已发布系统中更换编码器会产生什么影响，而不是普遍而言哪种冻结编码器最优；我们将在第五节 E 讨论这一问题。

IV. 实验设置

A. 数据与协议

所有系统均使用 ASVspoof 2019 LA (LA19)[25] 进行训练，不采用数据增强，与基线不使用数据增强的设置一致。我们在五个与训练集距离逐渐增大的基准测试上进行评估：LA19（同领域）；ASVspoof 2021 LA (LA21；近领域，存在编解码器和传输效应)；ASVspoof 2021 DF (DF21；经压缩的深度伪造音频) [3]；In-the-Wild (ITW；37.9 小时收集到的名人和政治人物录音) [4]；以及 ASVspoof 5 (ASV5；包含 16 种评测攻击的大规模众包数据) [26]。我们采用与基线相同的预处理 [5]：重采样至 16 kHz、预加重（0.97）、裁剪或填充至固定长度 64,600 个采样点，以及 25 ms/10 ms 分帧。保持前端一致，意味着系统间的差异反映的是编码器和后端，而非信号调理。所有系统，包括微调基线，均在同一条流水线中使用三个随机种子进行训练和评分。

B. 评估指标

我们报告三个随机种子（{42, 1234, 2025}）下汇总等错误率（%）的均值±标准差。每个随机种子的等错误率均采用单一的平局稳健约定；在文中引用时，95% 置信区间（针对类别分层的评估语句自助法百分位区间）用于量化运行内的评分不确定性。等错误率无阈值，并且可直接在全部五个基准测试之间进行比较。对于 ASV5（唯一采用标准化代价函数的数据集），我们还报告官方的反欺骗系统最小检测代价函数（$P_{\text{spoof}}=0.05$、$C_{\text{miss}}=1$、$C_{\text{fa}}=10$）。a-DCF 需要说话人验证分数，这超出了本反欺骗研究的范围。

C. 基线与参数统计

为在相同数据和评估流水线下比较冻结检测与全量微调，我们训练了一个流水线内微调基线：XLSR-53（315M 个参数[1]），解冻全部编码器权重并配备 AASIST后端，在 LA19 上训练，并按照与冻结系统相同的三个随机种子和约定进行评分。每个微调随机种子均依据最佳开发集等错误率进行选择（种子 42 为 0.12%），采用域外盲选规则（V-D 节）。我们的冻结融合头包含 390k 个可训练参数（MHFA单元）和 332k 个可训练参数（AASIST单元），而全量微调更新了 ∼316M 个参数，参数量减少了近三个数量级（约 800 倍）。我们明确说明，这是一项关于可训练参数的表述，而非关于模型大小或实测运行时间的表述：XEUS 本身就是一个大型编码器（577M 个参数），其推理开销与微调后的 XLSR-53系统相当，甚至更大。该结论关注的是必须进行优化的部分，而不是必须存储或运行的部分，这与 $\sim 10^5$参数的可训练头在冻结模型上报告的结果 [9]一致。

表 I 汇总等错误率（%），三个随机种子（{42, 1234, 2025}）下的均值±标准差，按冻结编码器和后端分类，所有系统均在同一流水线中训练和评分（越低越好）。微调基线（XLSR-53 未冻结 + AASIST）也使用三个随机种子，但未冻结。**Bold**/† 指四个析因设计系统：粗体标出每列的最佳结果，† 标出最佳冻结系统优于微调基线的域外基准测试（ASV5 在等错误率和最小检测代价函数方面；LA21 按配对自助法评估）。**mDCF**：ASVspoof 5 CM 官方最小检测代价函数。WavLM-Large+MHFA（316M）是容量匹配探针（第五节 E），不属于 2×2。

系统	可训练参数	LA19	LA21	DF21	ITW	ASV5	mDCF
微调（ft）	∼316M	2.54±1.71	11.75±0.91	6.34±3.62	24.37±6.37	27.81±2.15	0.77±0.05
XEUS +MHFA	0.39M	**2.12**±0.23	**10.93**±1.26†	**7.97**±0.58	**34.04**±2.98	**16.67**±0.78†	**0.48**±0.02†
XEUS + AASIST	0.33M	2.23±0.43	14.61±2.43	9.16±2.57	36.13±1.95	22.97±5.18	0.60±0.05
XLSR-53 + MHFA	0.39M	21.79±16.68	35.10±8.26	35.01±4.62	40.46±7.77	39.55±7.97	0.90±0.10
XLSR-53 + AASIST	0.33M	15.08±4.66	26.09±0.73	29.03±3.02	39.10±3.71	39.51±4.12	0.98±0.02
WavLM-Large + MHFA	0.39M	4.47±0.47	22.43±3.74	9.73±0.75	31.83±1.00	32.08±1.08	0.75±0.07

V. 结果与分析

A. 冻结编码器选择塑造跨数据集行为

表 I 报告了冻结析因设计在三个随机种子下的均值±标准差。编码器冻结时，XEUS 在每个单元中的平均等错误率都低于 XLSR-53。在 LA19、LA21、DF21 和 ASV5 上，这一差异较大且较为一致：在将后端固定为 MHFA 的情况下，冻结 XEUS 相较于冻结 XLSR-53，在 LA21（10.93 对 35.10）、DF21（7.97 对 35.01）和 ASV5（16.67 对 39.55）上的错误率降低约 58–77%。在 ITW 上，优势较小（34.04 对 40.46），且随机种子分布部分重叠，这与 ITW 对所有系统而言都是最困难的数据集之一相一致。在这一方案中，XEUS 更广泛、面向鲁棒性的预训练[6] 与一种能够迁移的固定表征相关，而冻结 XLSR-53 在很大程度上无法实现这一点。在匹配聚合下，该排序同样保持不变：在池化保持一致的情况下，XEUS 在每个基准上的平均等错误率仍低于 XLSR-53（V-B 节）。我们将其视为当前流水线中 XEUS 特有的一个实例，体现了一个正在形成的发现：编码器相较于后端，更能决定跨域行为 [9], [14], [27], ，而不是将其视为一项新原则。

为什么冻结的 XLSR-53 在不同领域中的表现如此不稳定? 冻结的 XLSR-53 在域内的等错误率较高，且强烈依赖随机种子（MHFA 的 LA19 为 21.79±16.68），读者可能会怀疑某个单元格配置错误。这种方差本身就是一项发现：在三个随机种子中，冻结的 XLSR-53+MHFA 要么收敛到理想的域内底线（LA19 10.40、14.03），要么表现接近随机猜测（40.95）；配置、数据和冻结特征均相同，唯一差异在于初始化。有两项观察表明，这种不稳定与冻结 XLSR 的设置有关，而非流水线普遍失效。首先，同一组权重在解冻 后，LA19 上的等错误率达到 0.85%（三个随机种子中的最佳值；三个随机种子的均值为 2.54，见表 I），说明该架构、数据和训练器都具备相应能力。其次，冻结的 XEUS 经由完全相同的融合、投影层和后端，能够稳定训练（LA192.12±0.23）。因此，我们报告所有系统的多随机种子均值±标准差，而非单次结果；该配置的单个随机种子结果处于这种不稳定

[1]315M（仅编码器）； ∼316M（包括 AASIST 头部）。

表 II 匹配聚合消融实验（MHFA 后端；在同一训练器上另行进行的三种子运行；随机种子的均值 ± 标准差）。对于每种编码器，只有 SSL 层聚合方式不同（最后一层与最后四层的均值）；其他设置均保持不变。**粗体** 标出每组匹配对中较低的等错误率（%）；XEUS 在全部十组比较中均更低。这是一项单独的单训练器研究；其绝对等错误率不可与表 I 直接比较，关注的比较是每行中 XEUS 与 XLSR 的相对排序。 {42, 1234, 2025}

	最终层		最后四层的均值	
数据集	XEUS	XLSR-53	XEUS	XLSR-53
LA19	**1.96**±0.64	31.17±10.55	**1.50**±0.24	27.61±15.83
LA21	**12.16**±0.20	44.02±4.35	**12.04**±0.51	37.79±8.68
DF21	**8.24**±0.63	39.87±4.28	**8.33**±0.65	35.89±3.86
ITW	**27.87**±4.43	52.41±3.50	**31.27**±0.79	44.46±3.12
ASV5	**18.28**±1.45	44.81±3.82	**23.46**±3.49	37.10±3.88

域外则截然不同：冻结的 XLSR-53 表现始终很差。即使是其表现最好的随机种子，在 LA21、DF21 和 ASV5 上的结果仍远逊于冻结的 XEUS，因此那里的编码器排序并非域内不稳定性的假象。

B. 匹配聚合下编码器排序依然成立

主要析因设计为每个编码器采用其最佳实践聚合方式（第 III-C节），因此 XEUS 的优势原则上可能来自编码器本身，也可能仅仅来自池化方式。为区分这两者，我们重新训练了匹配的 MHFA 单元：对于每个编码器，仅 改变聚合方式（最后一层与最后四层的均值），其他条件均保持不变，并使用相同的三个随机种子（表 II）。在全部十组匹配的 XEUS 与 XLSR-53 对比中（五个数据集 × 两种聚合方式），无论按三个随机种子的均值比较，还是依据互不重叠的 95%置信区间比较，XEUS 的等错误率都更低。因此，在匹配聚合下，编码器效应依然成立。

聚合是二阶效应，其方向在不同编码器之间并不一致：对最后四层取平均会降低 XLSR-53 在全部五项基准测试中的平均等错误率，而对 XEUS 而言，该效应较小且取决于基准测试。在领域（LA19）中，聚合有所帮助；在 LA21 和 DF21 中，其影响落在随机种子噪声范围内；而在两个差异较大的集合（ITW、

ASV5）中，最后一层表现更佳。一个能使某个编码器的表现整体发生变化、却只能不一致地影响另一个编码器的因素，不可能产生 XEUS 始终优于 XLSR 的统一排序；在两种聚合方案下的全部十个匹配单元中，这一排序均成立。

C. 编码器与后端的交互

在域内，对于冻结的 XEUS，两个后端的表现相当（ MHFA 2.12 对 AASIST 2.23，处于随机种子噪声范围内）；而对于冻结的 XLSR- 53，AASIST 是更稳定的选择（15.08±4.66 对 MHFA 21.79±16.68），因此域内后端排名已经取决于编码器。域外的这种交互更加明显。对于冻结的 XEUS，MHFA 在每个数据集上都是更优的后端（LA21 10.93 对 14.61；DF21 7.97 对 9.16；ITW 34.04 对 36.13；ASV5 16.67 对 22.97）；而对于冻结的 XLSR-53，AASIST 在每个数据集上都不逊于或优于 MHFA（LA21 26.09 对 35.10；DF21 29.03 对 35.01；ITW 39.10 对 40.46；ASV5 39.51 对 39.55）。因此，哪种后端效果最佳取决于其所读取的编码器。一种务实的解读是，MHFA 能够高效利用强大的固定表征，而 AASIST 更重的关系建模既能从较弱的表征中挽救更多信号，又能提高其稳定性：高方差单元是 XLSR-53+MHFA （域内标准差为 16.68 ），而将较弱的编码器与 AASIST 配对后，可将该标准差降至不到三分之一（4.66）。WeDefense 工具包[11]，报告称，在采用更新的前端时，MHFA 在 PartialSpoof（0.80% 对 1.04%）和 ASVspoof 5（8.08% 对 17.11%）上的表现优于 AASIST，这印证了 MHFA 与能力较强的编码器配对效果良好；MHFA 在自监督学习模型和任务之间的迁移能力 [28]也提供了进一步佐证。这意味着“最佳后端”并非与编码器无关；基线中“不同的后端分类器”这一未来工作项目 [5] 并未预见到这种交互，而这也与大规模证据相一致：经过组合的帕累托最优检测器子集优于朴素集成 [29]。

D. 冻结与全量微调：稳定性及局部反转

流水线内微调基线（XLSR-53 全部解冻的 + AASIST，三个随机种子）使我们能够在同一流水线和评分规则下比较冻结检测与全量微调。域内，表现最佳的微调随机种子在 LA19 上达到 0.85，但微调本身的随机种子稳定性较差（2.54±1.71），因此其三个随机种子的均值与冻结 XEUS+MHFA 的 2.12±0.23 相当，而非明显更优。域外的情况则好坏参半，且主要受微调的随机种子方差影响。在 ASVspoof 5 上，冻结的 XEUS+MHFA 明显优于拥有 316M 个参数的微调模型（16.67±0.78 对比 27.81±2.15；区间不重叠，差距达 11 个百分点）；在官方 ASV5 最小检测代价函数上也保持这一优势（0.48±0.02 对比 0.77±0.05），在 LA21 上则略占优势（10.93 对比 11.75；配对自助法，微调−冻结等错误率， $+0.82$ $[+0.64, +1.01]$，不包含零）。在两个中等距离的数据集上，微调模型表现更好：在 DF21 上，配对自助法结果为（-1.63 $[-1.88, -1.43]$），尽管其随机种子标准差大得多（3.62 对比 0.58）；在 In-the-Wild 上优势更明显（24.37±6.37 对比 34.04±2.98； -9.67 $[-10.14, -9.24]$）。关键在于，微调模型在域外的随机种子稳定性明显较差：三个随机种子下，其 DF21 结果范围为 4.04–10.52，ITW 结果范围为 18.93–31.37，因此即使其均值更优，单个微调模型的域外错误率仍难以预测。相比之下，冻结系统在 DF21、ITW 和 ASV5 上观测到的随机种子方差更低（LA21 上则相当，1.26 对比 0.91），并且在规模最大的基准上领先，同时更新的参数少得多。

运行内的不稳定性会加剧这一问题：对 seed-42 运行在第 5/10/15 轮进行探测（开发集等错误率分别为 0.30/0.12/0.12；我们报告第 10 轮的结果，因为它是开发集等错误率达到最佳值的最早轮次），域外错误率与开发集等错误率并不同步，因此，在这种情况下，单个随机种子、单个检查点的域外数值并不可靠。我们未能复现原始方案所报告的数值 [5], ，这些数值是在不同的流水线下获得的。我们的主张是基于同一流水线的比较。可以将其视为一种交换比率：由参数量不足 0.4M 的预测头读取的冻结表示，更新的参数约比 316M 参数的全量微调少 800 倍，并且在大多数域外数据集上对随机种子更稳定。它在 ASV5 和 LA21 上表现更佳，在 DF21 和 In-the-Wild 上落后，而在三随机种子均值上与其领域表现相当。冻结在域外可能与微调相媲美，这一点已有零星报道 [30]；我们的贡献在于展示了基于同一流水线、覆盖范围广泛的编码器，并刻画了随机种子稳定性。

E. 范围：容量与比较器的选择

以下两点限制了我们对编码器的解读。首先，在我们的冻结编码器对中，规模、多语言覆盖范围和预训练目标彼此混杂：XEUS（577M、4,057 种语言、去混响目标 [6]）比 XLSR-53（315M、53 种语言）更大、覆盖更广，因此匹配聚合消融实验表明，无论采用何种池化方式，排序都保持不变，但无法区分这三个维度。为探究 XLSR-53 的较弱结果是否只是使用 315M 冻结编码器的一般性后果，我们在析因设计之外加入了一个冻结的 WavLM Large [31]+MHFA 单元，作为容量匹配的诊断项（316M，与 XLSR-53 的 315M 匹配，采用相同流水线）。在容量匹配的情况下，WavLM Large 在域外以较大幅度优于 XLSR-53（LA21 为 22.43 对 35.10，DF21 为 9.73 对 35.01，ASV5 为 32.08 对 39.55），并且在域内训练稳定（LA19 为 4.47±0.47 对 21.79±16.68）。因此，315M 规模的冻结特征并不必然产生 XLSR-53 级别的性能，参数量本身也无法解释 XLSR-53 相对于 WavLM 的较弱表现。这与近期证据一致，即预训练的广度可能比原始参数量更为重要 [27]。不过，WavLM Large 在大多数数据集上仍未达到 XEUS（XEUS 在 LA19、LA21、DF21 和 ASV5 上胜出）。由于 XEUS 也更大（577M 对 316M），XEUS 相对于 WavLM 的优势仍混杂于规模和 XEUS 的预训练因素之间，我们不声称能够将二者区分开来。WavLM Large 在 ITW 上略胜 XEUS（31.83 对 34.04），因此，更广泛的覆盖范围并不能保证在此设置下取得更好的性能。ITW 仍是我们的冻结系统最困难的设置之一，也是微调基线优势最大的域外数据集。对于冻结编码器而言，这反映的是更难的攻击，还是仅仅不同类型的攻击 [23] ，目前尚无定论；而真实环境中的鲁棒性仍是整个领域尚未解决的问题。

VI. 讨论与局限性

预训练/评估溯源。若干基准测试与我们的编码器预训练语料库具有共同的溯源：LA19/LA21 以及 DF21 的很大一部分源自 VCTK，而 ASV5 则来自 MLS；这两个语料库都出现在 XEUS 的预训练数据中。对于这些数据集，我们无法完全将由表征驱动的行为与这种数据接触区分开来，因此将我们的论断限定在固定的下游方案上：更换冻结编码器会导致跨数据集性能出现显著且可重复的变化。两点观察可以为这一担忧提供信息，但无法消除它。首先，XLSR-53 本身也是在 MLS 上预训练的（其预训练数据以 MLS 为主），但在 ASV5 上比 XEUS 低 23 个等错误率点（16.67 对 39.55）。由于两个编码器都接触过 MLS，仅凭语料库存在性不足以解释它们之间的排序。不过，这一比较无法量化或排除数据重叠的影响。其次，从未见过 VCTK 的 WavLM Large 是一个容量匹配的编码器，但在 LA21 和 DF21 上仍以较大幅度超过 XLSR-53（22.43 对 35.10；9.73 对 35.01），因此，在源自 VCTK 的数据集上击败 XLSR-53 并不需要接触 VCTK。该优势在不包含 VCTK 的 ASV5 上同样成立。这些观察削弱了简单的“存在或不存在”解释，但并未估计数据重叠对 XEUS 性能的贡献。因此，对于存在溯源重叠的数据集，我们相应地限制对泛化能力的表述力度，而在固定下游方案内的刻画仍然成立。要隔离数据接触所产生的残余贡献，需要溯源清晰的基准测试，而这类基准目前仍然稀缺。

编码器与聚合、规模及目标。除编码器身份外，析因设计还改变了聚合方式、训练轮数和调度方案。第五节 B 对聚合因素进行了考察，而第五节 E 中容量匹配的 WavLMLarge 比较则部分探究了规模因素，使多语言覆盖与去混响目标彼此纠缠——对这些因素进行全面扫描将是未来的工作。

流水线、随机种子与增强。所有系统均在同一流水线中使用三个随机种子进行训练和评分。冻结的 XLSR-53 在域内以及微调后的域外表现均存在随机种子不稳定性，因此，对于二者而言，多随机种子报告都是必要的，而非装饰性的。遵循基线设置，我们不使用增强，仅在 LA19 上进行训练。RawBoost 风格增强 [32] 以及更广泛的数据很可能会同时帮助冻结和微调系统提升 ITW 与 ASV5 的表现。这种共享的训练设置，而非训练不足，解释了为什么我们的 XLSR-53 系统高于外部报告中的某些水平；我们的结论仅适用于这一流水线内部。

韵律。冻结与冻结的结果与韵律动机一致，但该设计无法将性能提升具体归因于韵律，而非更广泛的预训练，因此我们将其视为提示性证据，而非已得到证实的机制。

VII. 结论

冻结自监督学习编码器，将深度伪造检测转变为一个特征刻画问题：跨域行为从何而来？在这一方案中，这种行为在很大程度上取决于冻结编码器。替换编码器后，在 MHFA 后端上，LA21、DF21 和 ASV5 的跨数据集错误率降低了超过一半，而且在匹配聚合消融实验中，这一排序依然成立。容量匹配的 WavLM Large 诊断结果表明，XLSR-53 表现较弱，不能仅用参数量来解释。我们的设计无法区分更广的覆盖范围与去混响目标所产生的影响；此外，由于若干基准测试与编码器预训练所用数据具有共同来源，我们将结论限定于这一方案，而不声称实现了分布泛化。部分差异源自配对关系：后端的排名取决于其所读取的编码器，因此“最佳后端”并非可迁移的结论。微调也并非总能带来一致收益：与同一流水线中的三随机种子 316M 全量微调相比，由参数量不足 0.4M 的分类头读取的冻结广覆盖编码器，在同域表现相当，在大多数域外数据集上的观测种子间差异更小，并且在 LA21 和 ASVspoof 5 上表现更优；而微调方案则在 DF21 和 In-the-Wild 上胜出。

这些同样是部署方面的问题：在隐私敏感场景中投入使用的检测器，可能不会针对每种条件重新微调，因此，可信赖的部署要求我们了解哪些冻结表示能够迁移，以及如何高效地解读它们。特性分析是实现这一目标的基础，并非对性能的退让。

致谢

本研究部分使用了美国国家科学基金会 ACCESS 计划分配编号 CIS261055 所提供的计算资源。我们感谢 Sound Ethics 的 James O’Brien 指导我们。

OpenAI ChatGPT 和 Anthropic Claude 被用于对摘要、引言、相关工作、结果与讨论以及结论进行语言润色和改写。作者审阅并编辑了所有经 AI 辅助生成的文本，并对最终内容承担全部责任。

参考文献

[1] H. Tak、M. Todisco、X. Wang、J.-w. Jung、J. Yamagishi 和 N. Evans，“使用 wav2vec 2.0 和数据增强进行自动说话人验证欺骗与深度伪造检测”，载于*Odyssey*会议论文集，2022年，第112–119页。[2] X. Wang 和 J. Yamagishi，“探究用于语音欺骗对策的自监督前端”，载于*Odyssey*会议论文集，2022年，第100–106页。[3] X. Liu、X. Wang、M. Sahidullah、J. Patino、H. Delgado、T. Kinnunen、M. Todisco、J. Yamagishi、N. Evans、A. Nautsch 和 K. A. Lee，“ASVspoof 2021：迈向真实环境中的伪造语音与深度伪造语音检测”，*IEEE/ACM*《音频、语音与语言处理汇刊》，第31卷，第2507–2522页，2023年。[4] N. M. M¨uller、P. Czempin、F. Dieckmann、A. Froghyar 和 K. B¨ottinger，“音频深度伪造检测能否实现泛化？”，载于*Interspeech*会议论文集，2022年，第2783–2787页。[5] Y. El Kheir、A. Das、E. E. Erdogan、F. Ritter-Guti´errez、T. Polzehl 和 S. M¨oller，“两个视角，一个真相：用于鲁棒语音深度伪造检测的频谱特征与自监督特征融合”，载于*IEEE*音频与声学信号处理应用研讨会论文集（*WASPAA*），美国加利福尼亚州塔霍城，2025年，第1–5页，doi: 10.1109/WASPAA66052.2025.11230938。[6] W. Chen、W. Zhang、Y. Peng、X. Li、J. Tian、J. Shi、X. Chang、S. Maiti、K. Livescu 和 S. Watanabe，“面向数千种语言的鲁棒语音表征学习”，载于*EMNLP*会议论文集，2024年，第10205–10224页。[7] A. Conneau、A. Baevski、R. Collobert、A. Mohamed 和 M. Auli，“用于语音识别的无监督跨语言表征学习”，载于*Interspeech*会议论文集，2021年，第2426–2430页。

[8] M. Frohmann、G. Meseguer-Brocal、M. Schedl 和 E. V. Epure，“双关语：通过多视图融合实现基于音频的人工智能生成歌词的鲁棒检测”，载于 ACL Findings，2025年，第1914–1926页。 [9] P. Serrano、R. Duroselle、F. Angulo、J.-F. Bonastre 和 O. Boeffard，“通过层选择与基于自监督学习的反制措施融合，改进域外音频深度伪造检测”，arXiv：2509.12003，2025年。 [10] D. Afchar、G. Meseguer-Brocal 和 R. Hennequin，“音乐深度伪造检测看似容易，实则困难”，arXiv：2405.04181，2024年。 [11] L. Zhang、J. Rohdin、X. Wang、J. Peng、T. Liu、Y. Zhang、H.-T. Luong、S. Wang、C. Liang、A. Silnova 和 N. Evans，“WeDefense：用于防御伪造音频的工具包”，arXiv：2601.15240，2026年。 [12] E. J. Hu、Y. Shen、P. Wallis、Z. Allen-Zhu、Y. Li、S. Wang、L. Wang 和 W. Chen，“LoRA：大型语言模型的低秩适配”，载于 ICLR会议论文集，2022年。 [13] K. Borodin、V. Kudryavtsev、D. Korzh、A. Efimenko、G. Mkrtchian、M. Gorodnichev 和 O. Y. Rogov，“AASIST3：用于 ASVspoof 2024挑战赛的 KAN增强型AASIST语音深度伪造检测：使用自监督特征与额外正则化”，载于 ASVspoof 2024研讨会（自动说话人验证欺骗反制措施），希腊科斯岛，2024年，第48–55页，doi：10.21437/ASVspoof.2024-8。 [14] H. Ali、N. S. Adupa、S. Subramani 和 H. Malik，“用于音频深度伪造检测的自监督语音模型 SUPERB 风格基准”，载于 IEEE 国际声学、语音与信号处理会议（ICASSP），2026年，arXiv：2603.01482。 [15] J.-w. Jung、H.-S. Heo、H. Tak、H.-j. Shim、J. S. Chung、B.-J. Lee、H.- J. Yu 和 N. Evans，“AASIST：使用集成时频图注意力网络的音频反欺骗”，载于 IEEE 国际声学、语音与信号处理会议，2022年，第6367–6371页。 [16] J. Peng、O. Plchot、T. Stafylakis、L. Moˇsner、L. Burget 和 J. ˇCernock´y，“一种基于注意力的后端，可对 Transformer 模型进行高效微调以实现说话人验证”，载于 IEEE SLT，2022年，第555–562页。 [17] A. Pasad、J.-C. Chou 和 K. Livescu，“自监督语音表征模型的逐层分析”，载于 IEEE ASRU，2021年，第914–921页。 [18] Y. El Kheir、Y. Samih、S. Maharjan、T. Polzehl 和 S. M¨oller，“用于音频深度伪造检测的自监督学习模型全面逐层分析”，载于 NAACL Findings，2025年，第4070–4082页。 [19] T. Liu、D.-T. Truong、R. K. Das、K. A. Lee 和 H. Li，“Nes2Net：由基础模型驱动的语音反欺骗轻量级嵌套架构”，《IEEE信息取证与安全汇刊》，第20卷，第12005–12018页，2025年，doi：10.1109/TIFS.2025.3626963。 [20] A.-T. Dao、D. Matrouf、M. Rouvier 和 N. Evans，“增强基于自监督学习的反欺骗模型中的多语料库训练：域不变特征提取”，arXiv：2603.18657，2026年。 [21] U. Thani、G. Singh 和 P. Singh，“通过模块化统计变换实现音频深度伪造检测的无监督域适应”，载于 IEEE/CVF 计算机视觉与模式识别会议研讨会（CVPRW），2026年，第11031–11039页。 [22] D. Li、J. Xue、Y. Ren、Z. Yi、Y. Huang、G. Feng 和 Y. Chai，“当前语音深度伪造检测方法对真实世界的泛化能力如何？”，arXiv：2603.05852，2026年。 [23] N. M. M¨uller、N. Evans、H. Tak、P. Sperl 和 K. B¨ottinger，“更难还是不同？理解音频深度伪造检测的泛化”，载于 Interspeech会议论文集，希腊科斯岛，2024年，第2705–2709页，doi：10.21437/Interspeech.2024-247。 [24] M. Todisco、H. Delgado 和 N. Evans，“恒定 Q 倒谱系数：一种用于自动说话人验证的欺骗对抗措施”，《计算机语音与语言》，第45卷，第516–535页，2017年。 [25] X. Wang 等，“ASVspoof 2019：合成、转换与重放语音的大规模公共数据库”，《计算机语音与语言》，第64卷，2020年，文章编号101114，doi：10.1016/j.csl.2020.101114。 [26] X. Wang、H. Delgado、H. Tak、J.-w. Jung、H.-j. Shim、M. Todisco、I. Kukanov、X. Liu、M. Sahidullah、T. H. Kinnunen、N. Evans、K. A. Lee 和 J. Yamagishi，“ASVspoof 5：大规模众包语音数据、深度伪造与对抗攻击”，载于 ASVspoof 2024研讨会（自动说话人验证欺骗反制措施），希腊科斯岛，2024年，第1–8页，doi：10.21437/ASVspoof.2024-1。 [27] A. Kulkarni、S. Dowerah、A. Kulkarni、T. Alum¨ae 和 M. Magi- mai Doss，“紧凑型自监督学习骨干网络对音频深度伪造检测是否重要？使用 RAPTOR 的受控研究”，2026年，arXiv：2603.06164。 [28] J. Peng、L. Moˇsner、L. Zhang、O. Plchot、T. Stafylakis、L. Burget 和 J. ˇCernock´y，“CA-MHFA：一种上下文感知多头因子化用于基于自监督学习的说话人验证的注意力池化，”载于*IEEE* 国际声学、语音与信号处理会议论文集，2025年，第1–5页。 [29] V. Stanˇek、M. Pereˇs´ıni、L. Sekanina、A. Firc 和 K. Malinka，“语音深度伪造检测器的进化多目标融合”，载于*IEEE* 进化计算大会（*CEC*）论文集，2026年，arXiv:2604.01330。 [30] I. Viakhirev、D. Sirota、A. Smirnov 和 K. Borodin，“迈向可扩展的 AASIST：改进用于语音深度伪造检测的图注意力”，2025年，arXiv:2507.11777。 [31] S. Chen 等，“WavLM：面向全栈语音处理的大规模自监督预训练”，*IEEE* 信号处理专题期刊，第16卷，第6期，第1505–1518页，2022年10月。 [32] H. Tak、M. R. Kamble、J. Patino、M. Todisco 和 N. Evans，“Raw-Boost：一种应用于自动说话人验证反欺骗的原始数据增强方法”，载于*IEEE* 国际声学、语音与信号处理会议论文集，2022年，第6382–6386页。