arXiv:2303.03591v1 [cs.SD] 7 Mar 2023

通过批次嵌入协方差正则化与常数 Q 变换学习广义音频表征的方法

Ankit Shah
语言技术研究所 卡内基梅隆大学
匹兹堡，宾夕法尼亚州 15213
aps1@andrew.cmu.edu

Shuyi Chen
海因茨信息学院 卡内基梅隆大学 匹兹堡，宾夕法尼亚州 15213
shuyic@andrew.cmu.edu

Kejun Zhou
海因茨信息学院 卡内基梅隆大学 匹兹堡，PA 15213
kejunz@andrew.cmu.edu

Yue Chen
海因茨信息学院 卡内基梅隆大学 匹兹堡，PA 15213
yuechen2@andrew.cmu.edu

Bhiksha Raj
语言技术研究所 卡内基梅隆研究院 匹兹堡，PA，15213
bhiksha@cs.cmu.edu

摘要

在包括音频任务在内的许多应用场景中，通用嵌入对于少样本乃至零样本学习都极具价值。为了更好地理解表征，我们对 HEAR 2021 提交结果进行了全面的误差分析和可视化。受该分析启发，本文尝试了不同的前端音频预处理方法，包括常数 Q 变换（CQT）和短时傅里叶变换（STFT），并提出批量嵌入协方差正则化（BECR）项，以更全面地模拟人类听觉系统接收到的频率信息。我们在 HEAR 2021 任务套件上测试了这些模型，该套件涵盖了广泛的任务类别。初步结果表明：（1）所提出的 BECR 能够使测试集中的嵌入分布更加分散；（2）BECR 无需增加计算复杂度即可提升 PaSST 模型的性能；（3）在我们测试的所有任务中，STFT 预处理均优于 CQT。

GitHub： https://github.com/ankitshah009/general_audio_embedding_hear_2021

1 引言

在音频数据集领域，通用表征学习仍是一个悬而未决的问题。因此，提出了 2021年音频表征整体评估（HEAR 2021）挑战，旨在深入了解不同广义音频表征模型的长期表现 [1]。这项挑战要求训练一个音频表征模型，使其具备足够的灵活性，能够表征未见过的音频数据集。通过在模型的嵌入输出上训练并测试一个浅层网络，对这些表征进行评估。HEAR2021 的端到端流程如图 1 所示。

受挑战赛结果的启发，本文首先比较短时傅里叶变换（STFT）与常数 Q 变换（CQT）；在挑战赛中，没有任何团队将 CQT 用作音频预处理方法。其次，基于对 HEAR 2021 提交结果的全面误差分析和可视化，我们提出了批量嵌入协方差正则化（BECR），

草稿。

这是一种正则项，利用基尼指数衡量某项训练任务中嵌入的协方差矩阵特征值的统计离散程度。具体而言，它鼓励将特定预训练任务的表征投影到其所有特征向量上，并使其尽可能均匀地分散。因此，其目标是在仅使用特定领域的单个数据集进行训练时，学习能够在低维空间中更具通用性的深度表征网络。我们还提出了一种优化的实现算法，以降低时间复杂度。

我们在四项 HEAR 2021 任务上测试了这两种方案，并与基线模型进行了比较。这些任务涵盖语音、音乐和广义音频三个领域。结果表明，BECR 在所有任务中都提升了 PaSST 基线模型的表现，而使用 CQT 训练得到的结果则不如 Mel STFT 模型。

2 相关工作

2.1 音频数据预处理技术

2.1.1 短时傅里叶变换（STFT）和梅尔频谱图

解决通用表征学习挑战的一种更好方法，是根据领域专业知识，针对不同任务应用不同的手工设计变换，例如使用短时傅里叶变换（STFT） [1]。STFT 是一种强大的音频信号处理工具，可用于许多任务。它描述每个信号随时间和频率变化的复振幅，并定义了一类有价值的时频分布。然而，STFT 也有其缺点，例如时频分辨率能力有限。短窗难以描绘低频，而长窗则难以准确定位短脉冲的时间。[2]由于人类并非以线性刻度感知声音，因此提出了梅尔刻度，使音高上的等距间隔在人类听来也是等距的。梅尔频谱图将赫兹值转换为梅尔刻度。与 STFT 相比，这种变换更能模拟人类听觉 [3]。

2.1.2 常数 Q 变换（CQT）

另一种适用于预处理音乐、人声及其他声音变化数据的方法是常数 Q 变换。1991 年，Brown 提出 CQT，通过使用具有固定品质因数 Q 的变换来模拟人类听觉系统 [4]。常数 Q 变换与 STFT 有几方面不同。常数 Q 变换的频率箱按对数间隔分布，而 STFT 的频率分量则是线性的。此外，常数 Q 变换的频率箱宽度按八度变化，而非固定不变。由于变换的输出实际上是相对于对数频率的振幅/相位，因此只需较少的频率箱即可有效覆盖给定范围 [3]。因此，有些人认为，常数 Q 变换能更好地描述人类听觉系统接收到的内容，因而更适用于音乐领域 [5]。

2.2 机器学习中的基尼指数

基尼指数是一种数据纯度度量。基尼指数较小，表明编码或信号的纯度较高。基尼指数已广泛应用于机器学习。例如，Randall [6] 提出了神经决策树（NDT），在实践中结合基尼指数与神经网络，基于决策树和多层感知机构建该模型。Park [7] 还提出了一种深度学习模型，使用基尼指数算法从数据集中提取特征。

在分析 HEAR2021 结果时，我们意识到需要一种汇总统计量，用来描述嵌入矩阵在评估数据集上的整体几何特性——具体来说，就是不同任务的嵌入分布得有多分散。因此，我们想到将基尼指数应用于嵌入矩阵的归一化特征值，并将其用作正则化项。我们认为，这项工作首次在音频任务中采用了这种定义的基尼指数。

3 方法

本研究的端到端流程如图 1 所示。我们首先比较两种预处理方法的效果。其次，我们尝试一种新颖的基于基尼指数的正则化方法，以提高模型的通用性。利用所得模型在

HEAR2021 数据集中的多种未知数据集上生成嵌入，并用这些嵌入训练浅层 MLP 层，以获得最终的评估分数。

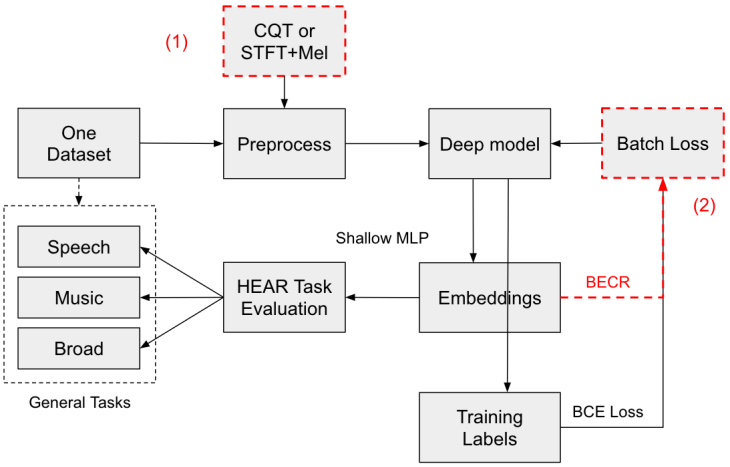


图1：HEAR2021 的端到端训练与评估流程。在本研究中，我们对其进行了两项修改：(1) 比较 CQT 和 STFT 的效果；(2) 在训练过程中设计了一个正则化项。

3.1 预处理方法

短时傅里叶变换（STFT）首先将较长的录音信号在时域划分为长度相等的短片段。然后，对每个片段进行傅里叶变换，生成多个频谱。离散 STFT 可表示为：

$$X(m,\omega) = \sum_{n=-\infty}^{\infty} x[n]\omega[n-m]e^{-j\omega n} \tag{1}$$

CQT 变换模拟人类听觉系统，在较低频率处具有更高的频谱分辨率 [8]。离散 CQT 可表示为：

$$X[k] = \frac{1}{N[k]} \sum_{n=0}^{N[k]-1} W[k,n]x[n]e^{\frac{-j2\pi Qn}{N[k]}} \tag{2}$$

3.2 基线架构

我们选择 PaSST 模型作为基线结构 [9]，该模型在 HEAR2021挑战赛的 19 项任务（其中 14 项为秘密任务）中取得了总体最佳结果 [1]。PaSST 是最先进的基于Transformer的音频模型，与其他基于CNN的模型相比，它能够以更低的内存占用和时间复杂度取得最先进的结果 [9]。如图 2 所示，模型的输入是由预处理方法生成的音频频谱图。在第 1 部分，输入经过分块提取和线性投影。在第 2 部分，添加频率和时间位置编码。随后进行 Patchout 操作。Patchout 的思路是鼓励 Transformer 使用不完整的序列进行分类，其效果类似于 Dropout。最后，将序列展平并输入自注意力层。最后，分类器MLP层对分类标记进行处理并生成预测结果 [9]。

3.3 批次嵌入协方差正则化（BECR）

3.3.1 低维表示投影分析

通过分析不同任务中顶尖模型的嵌入，我们观察到，对于同一任务，在下游任务中表现更好的模型通常具有更高的嵌入

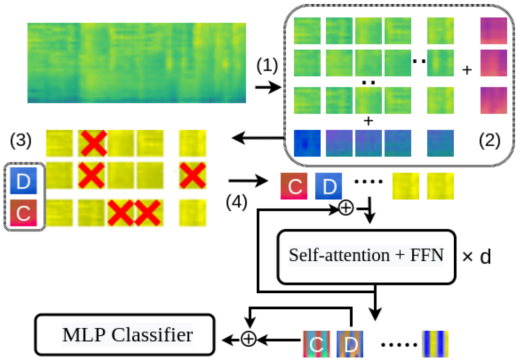
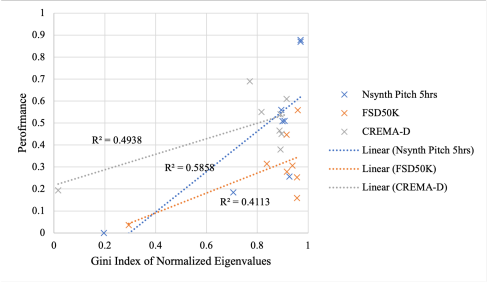
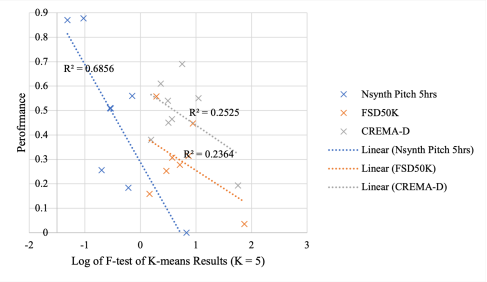


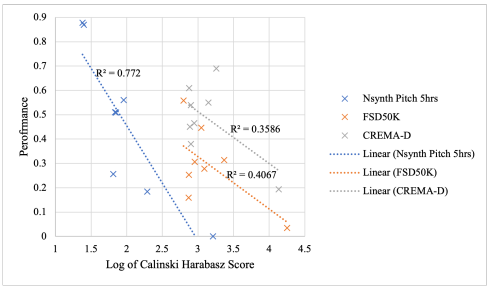
图 2：Patchout Transformer（PaSST）架构 [9]。



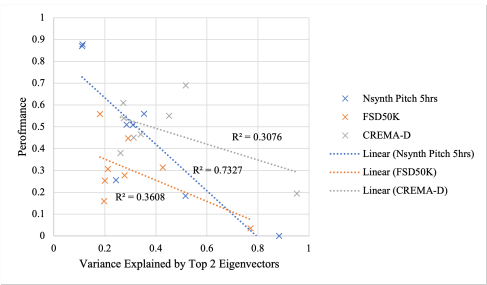
(a) 归一化特征值的基尼指数



(b) K均值结果的 F检验



(c) K均值结果的 Calinski-Harabasz分数



(d) 前两个主向量解释的方差

图 3：嵌入性能和离散度指标汇总。每个数据点代表一个提交模型或复现的HEAR2021模型。NsynthPitch、FSD50K 和 CREMA-D 分别对应音乐、广泛音频和语音任务。

离散度，例如由主成分解释的方差较低，且 K 均值 F 检验分数较低。总结见图 3。因此，我们推测，模型生成的嵌入具有较高的离散度，有助于下游任务。

3.3.2 BECR设计

由于我们将训练过程限制为仅使用一个任务数据集，因此协方差矩阵的特征值适合用于模拟嵌入投影到不同维度时，在不同测试任务上的方差。因此，归一化特征值的基尼指数是一种汇总统计量，用于描述嵌入在特征空间中的分散程度是否均匀。由此，我们将 BECR 定义如下：

对于具有 D 个输出的嵌入层，$\mathbf{K}$ 是每个批次嵌入 $f_\theta(\mathbf{X}_i)$ 的 D 维协方差矩阵，

$$\mathbf{K}(f_\theta(\mathbf{X}_i)) = (f_\theta(\mathbf{X}_i) - \mathrm{E}[f_\theta(\mathbf{X}_i)])(f_\theta(\mathbf{X}_i) - \mathrm{E}[f_\theta(\mathbf{X}_i)])^{\mathrm{T}} \tag{3}$$

$\mathbf{K}$ 始终是半正定的，其特征值均为实数且非负。 $\mathcal{G}$ 将基尼指数的定义应用于$\mathbf{K}$的归一化特征值,

$$\mathcal{G}(f_\theta(\mathbf{X}_i)) = 1 - \sum_{i=1}^{D} (\frac{\lambda_i(\mathbf{K}(f_\theta(\mathbf{X}_i)))}{\sum_{i=1}^{D} \lambda_i(\mathbf{K}(f_\theta(\mathbf{X}_i)))})^2 \tag{4}$$

$\mathcal{R}$ 是所提出的正则化项,

$$\mathcal{R}(\mathbf{X}_i, \theta) = \max(0, \epsilon - \mathcal{G}(f_\theta(\mathbf{X}_i)))^2 \tag{5}$$

其中， ϵ是一个超参数，用于定义产生损失时基尼指数的上界。
最后，总损失定义为

$$\mathcal{L}'(\mathbf{X}_i, \theta) = (1 - \lambda)\mathcal{L}(\mathbf{X}_i, \theta) + \lambda\mathcal{R}(\mathbf{X}_i, \theta) \tag{6}$$

在我们的设置中，原始损失 $\mathcal{L}$ 是二元交叉熵损失，其取值范围为 [0, 1]。通过添加正则化项，可以鼓励基尼指数增大，从而促使特征值分布更加均匀。此外，正则化项是添加到原始损失函数中的凸函数，具有理想的收敛性质。

3.3.3 BECR 的实现细节

设批量大小为 N，嵌入维度为 K。特征值分解算法的复杂度为$O（K^3）$。加上协方差矩阵的计算,每个批次增加的总复杂度为 $O（K^3 + K^2 * N）$；对于我们的情况，这显然不切实际，因为 N 约为 10，而 K 约为 1000。实验表明，在一台 RTX 3090 机器上，对 FSD50K 数据集的一个包含 10 个样本的批次进行训练并计算嵌入的特征值分解，大约需要 6 小时，比不使用该损失时长 18 倍。因此，我们提出了一种无需特征值分解的优化实现，其复杂度为 $O（K^2 * N）$。

回顾 $tr(A)、= \sum_i \lambda_i(A)$ 和 $tr(A^2)、= \sum_i \lambda_i(A)^2$。因此，公式 4 可简化为

$$\mathcal{G}(f_\theta(\mathbf{X}_i)) = 1 - \sum (\frac{\lambda_i}{\sum \lambda_i})^2 = 1 - \frac{\sum \lambda_i^2}{(\sum \lambda_i)^2} = 1 - \frac{tr(\mathbf{K}(f_\theta(\mathbf{X}_i))^2)}{tr(\mathbf{K}(f_\theta(\mathbf{X}_i)))^2} \tag{7}$$

通过实验发现，该实现方法的速度与原始损失的计算速度相近。如表 3 所示，简化后的 BECR 在训练中耗时 33 小时，而原始损失耗时 36 小时。因此，我们可以在几乎不增加额外复杂度的情况下应用 BECR。

4 实验评估

4.1 数据集与评估指标

我们在三类数据集上评估模型的性能：音乐、语音和广义声音。数据集和评估指标概要见表 1。

对于音乐，我们使用包含 305,979 个音乐音符的 NSynth Pitch，每个音符都具有独特的音高、音色和包络。对于来自商业采样库的 1,006 种乐器，数据集通过遍历标准 MIDI 钢琴的每个音高（21–108）以及五种不同力度（25、50、75、100、127），生成了时长为四秒的单声道 16kHz 音频片段。此任务旨在将乐器声音分类为 88 个音高之一 [10]。NSynth Pitch 任务使用音高准确率进行评估。

我们还使用京剧打击乐进行音乐任务评估。京剧打击乐器数据集基于京剧 [11] 的录音构建，包含六种主要打击乐器，可分为四大类：板鼓、铙钹、大锣和小锣。该数据集共有 236 段音频片段。京剧打击乐任务使用分类准确率进行评估。

对于语音，我们使用 CREMA-D 进行情绪识别 [12]。该数据集包含演员朗读句子的音频数据，句子分别表达六种不同的情绪之一（愤怒、厌恶、恐惧、快乐、中性和悲伤）。此任务旨在识别演员朗读句子时所处的情绪类型。使用分类准确率评估 CREMA-D 任务。

对于广义声音，我们使用 FSD50K。该数据集中的每段音频都使用环境音、语音和音乐领域中的 200 个类别中的一个或多个进行标注 [13]。该数据集包含超过 51K 段音频，总时长超过 100 小时，数据取自 AudioSet 本体。我们还使用 FSD50K 进行训练。使用平均精度均值（mAP）对 FSD50K 任务进行多标签评估。

任务名称	预测器类型	划分模式	评估指标
Nsynth 音高 5h	C	TVT	音高准确率
京剧	C	5 折	准确率
CREMA-D	C	5 折	准确率
FSD50K	L	TVT	mAP

表 1：从 HEAR 2021 [1] 中选取的四项评估任务概述。对于这四项任务，嵌入类型均为基于场景。预测器类型为多类别 (C) 或多标签 (L)。下游评估所用的划分方法为训练/验证/测试 (TVT) 或 K 折。

4.2 超参数调优

所提出的 BECR 涉及两个超参数，λ 和 ϵ。我们仅测试了 λ 为 0.05 和 0.10 的情况，因为在我们的实验中，$\lambda \leq 0.1$ 可确保 BECR 不会在最初几个训练周期中吞噬全部损失。

为确定 ϵ的搜索空间，我们观察到，在训练100个训练轮次后，FSD50K上原始PaSST的嵌入基尼指数为0.92（见表4）。此外，在初始批次中，实验显示基尼指数约为0.3-0.6。因此，如果将epsilon设置为小于0.6，则没有意义，因为这对最终输出几乎没有影响（见公式5）。所以，我们进行了 ϵ 大于0.7的实验。表2中的调参结果显示，$(\lambda,\ \epsilon) = (0.05,\ 0.7)$是最佳组合。

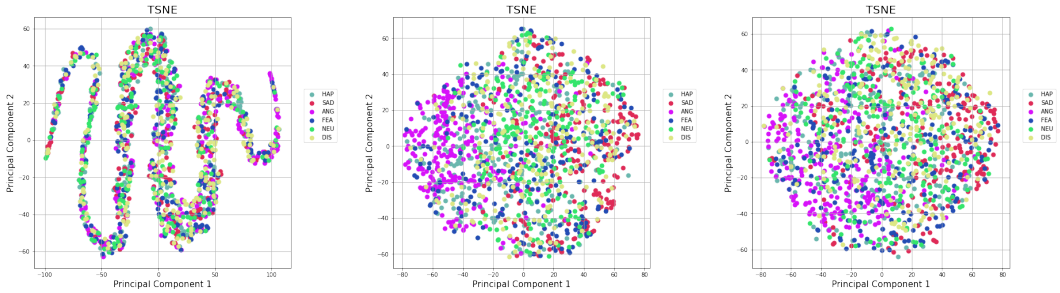
ϵ	λ	# 训练轮次	测试集每次-性能
0.7	0.05	50（19小时）	**30.7**
0.8	0.05	50（18小时）	24.4
0.9	0.05	50（18小时）	24.9
0.7	0.10	50（19小时）	25.6
0.8	0.10	50（18小时）	25.7
0.9	0.10	50（18小时）	29.8

表2：训练集（FSD50K）上的超参数调优结果

.

5 结果

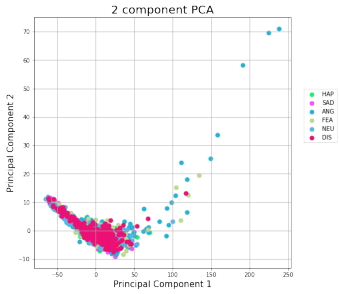
结果表明，在所有四项任务中，CQT预处理都不如 STFT+梅尔。除此之外，CQT变换的计算复杂度高于 STFT+梅尔，耗时超过原始 STFT+梅尔的两倍（在 FSD50K 上，批量大小为



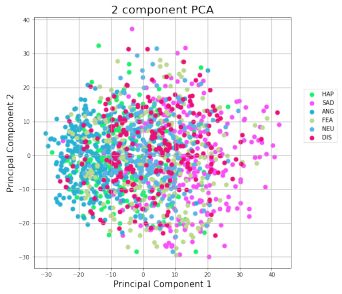
(a) 基于 CQT 的 PaSST 的 t-SNE 模型嵌入

(b) 基于 STFT 的 PaSST 的 t-SNE 模型嵌入

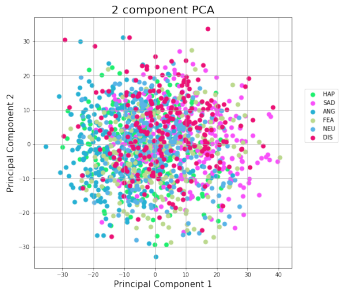
(c) 基于 STFT 的 PaSST 的 t-SNE 采用 BECR 的模型嵌入



(d) 基于 CQT 的 PaSST 模型嵌入的主成分分析



(e) 基于 STFT 的 PaSST 模型嵌入



(f) 基于 STFT 的 PaSST 的主成分分析 BECR 模型嵌入

图 4：使用主成分分析和 t-SNE 对嵌入进行投影。数据点根据真实标签着色。数据集为 CREMA-D。模型是在 FSD-50K 上训练的，而非 CREMA-D 数据集。

6). 所提出的 BECR 结合梅尔+STFT，其在所有四项任务中的表现均优于基线模型，且训练复杂度相近。

模型	**# 训练轮次**	**下游评估分数(%)** 京剧	Nsynth 音高 5	CREMA-D	FSD50K
STFT 梅尔+PaSST（基线）	100（33小时）	90.6	50.9	46.5	27.8
STFT 梅尔+PaSST（BECR）	100（36小时）	**92.7**	**51.2**	**47.6**	**36.8**
CQT+PaSST	50（50小时）	36.8	4.8	19.4	3.5
HEAR2021 [1]中的 CP-JKU PaSST	未知	96.6	25.6	61.0	55.8

Tabl表 3：模型性能总结。BECR 的 ϵ 和 λ 在第 4.2 节的训练集上进行了调优 .

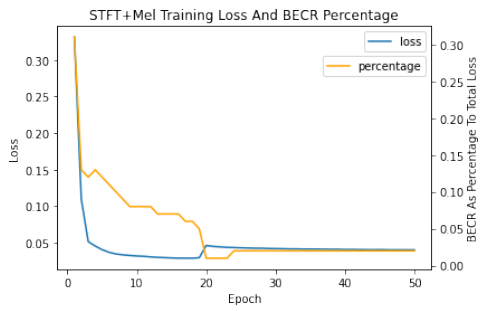
5.1 CQT 结果讨论

我们尝试寻找 CQT 的性能不如 STFT 的原因。

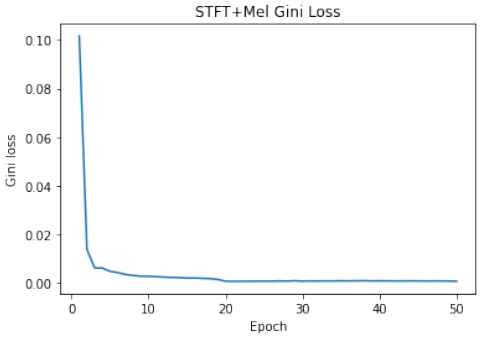
首先，通过比较基于 STFT 和 CQT 的 PaSST 模型经 PCA 和 t-SNE 降维后得到的嵌入，我们发现，CQT 的投影通常呈现线性不可分形状，而 STFT 的投影则似乎更像聚类。考虑到评估过程会在嵌入结果上构建一个两层多层感知器（见图1），因此可以合理推测，基于 CQT 的 PaSST 嵌入不可分，是其在下游任务中性能不佳的原因。

其次，基于 CQT 的 PaSST 的性能及其部分指标符合第 3.3.1 节所述的模式。与基于 STFT 的 PaSST 模型相比，基于 CQT 的 PaSST 嵌入结果在未见数据集上的分散程度较低。见表4。

第三，撇开实验结果不谈，我们认为，相对“良好”的预处理方法在一定程度上取决于模型的选择。PaSST 的原始实现采用短时傅里叶变换[9], ，因此 PaSST 模型使用 STFT 预处理方法时效果最佳，而非其他方法。

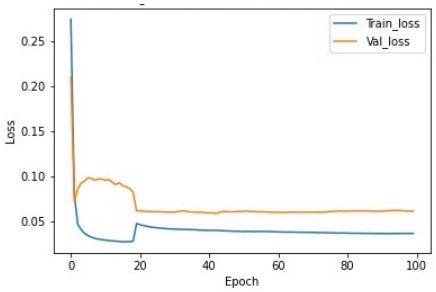


(a) 总损失下降及 BECR 占总损失的比例

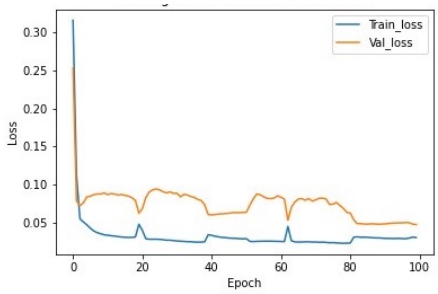


(b) BECR 项下降

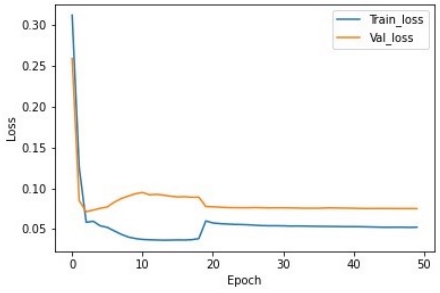
图 5：BECR 下降与总训练损失下降的比较（$\lambda = 0.05, \epsilon = 0.7$）



(a) 使用 STFT + 梅尔的 PaSST（基线）



(b) 使用 BECR 和 STFT +梅尔的 PaSST



（c）采用 CQT 的 PaSST

图 6：模型的训练损失下降情况

5.2 BECR 结果讨论

为验证 BECR 确实改善了基线 PaSST 模型，我们展示了正则化项在最初几个训练轮次后成功收敛至零的情况。见图 5。具体而言，BECR 项占总损失的比例稳步从 15% 降至 0%，即使在训练初期也不会吞噬总损失。因此，与基线模型相比，在 100 训练轮次的训练过程中，验证损失的下降并未受到显著影响（图 6）。

另一项证据是，在基线 PaSST 中加入 BECR 后，前几个特征向量所解释的方差减少，F 检验分数降低，归一化特征值的基尼指数升高，这正是 BECR 的预期效果（见表4）。这些变化表明，嵌入投影比以前分布得更加分散；根据我们在第 3.3.1 节中的分析，这会带来更好的性能。

数据集	模型	基尼指数	前 2 个特征值比率	性能
Nsynth Pitcht 5h	CQT+PaSST	0.20	0.88	0.048
	STFT+PaSST	0.90	0.31	0.509
	STFT+PaSST（BECR）	**0.91**	**0.29**	**0.512**
FSD50K	CQT+PaSST	0.29	0.77	0.035
	STFT+PaSST	0.92	0.28	0.278
	STFT+PaSST（BECR）	**0.94**	**0.21**	**0.368**
CREMA-D	CQT+PaSST	0.02	0.95	0.195
	STFT+PaSST	0.89	0.34	0.465
	STFT+PaSST（BECR）	**0.90**	**0.31**	**0.476**

表4：BECR 的效果。模型在 FSD50K 数据集上以批量大小 10 训练 100 个训练周期。

6 结论

本文证实，与 PaSST 模型搭配使用时，CQT 是不如 STFT+梅尔理想的音频预处理方法：它不仅会降低下游任务的性能，还会在将音频数据转换到频域时大幅增加计算复杂度。

我们还提出了批量嵌入协方差正则化（BECR），这是一种基于基尼指数的正则化项，旨在促进嵌入在其特征空间中的广泛分布，并为此提出了一种快速实现算法。我们在广泛的音频领域中使用最先进的音频模型 PaSST 对 BECR 进行了测试，包括音乐、语音和广义音频；与仅使用 PaSST、在相同数据集上以相近时长训练的情况相比，BECR 取得了更好的性能。其直观简洁、实现复杂度低，且在具有挑战性的未知测试任务中取得了令人鼓舞的结果，这些都表明 BECR 在通用音频表征学习方面具有良好的应用潜力。

我们还需要指出本工作的若干局限性。首先，不同模型适用于不同的预处理方法。因此，关于 CQT 优于 STFT 的结论仅限于使用 PaSST 模型的实验设置。其次，我们尚未验证 BECR 是否是一种能够推广到 PaSST 模型之外的通用技术。未来将 BECR 应用于其他常见基线模型会很有意义，例如 OpenL3 和 wav2vec2 模型，并观察其带来的差异。

致谢

感谢 Chaoran Zhang 和 Yuxiang Zhang 对这项工作提出的宝贵意见。

参考文献

[1] Joseph Turian、Jordie Shier、Humair Raj Khan、Bhiksha Raj、Björn W Schuller、Christian J Steinmetz、Colin Malloy、George Tzanetakis、Gissel Velarde、Kirk McNally 等人，《Hear：音频表征的整体评估》，载于*NeurIPS 2021* 竞赛与演示专题。PMLR，2022年，第125–145页。[2] 《局部时频分析与短时傅里叶变换》，https://www.math. ucdavis.edu/~strohmer/research/gabor/gaborintro/node3.html，2010，[在线；访问日期：5月 2, 2022,]日。[3] Ziqiang Shi、Huibin Lin、Liu Liu、Rujie Liu 和 Jiqing Han，《CQT 是否比 STFT 更适合单声道语音分离？一项实证研究》，*arXiv*预印本 *arXiv:1902.00631*，2019年。[4] DeLiang Wang 和 Guy J Brown，《计算听觉场景分析：原理、算法与应用》，Wiley-IEEE出版社，2006年。

[5] Christian Schörkhuber 和 Anssi Klapuri，《用于音乐处理的常数 Q 变换工具箱》，载于第七届声音与音乐计算会议，西班牙巴塞罗那，2010年，第3–64页。[6] Randall Balestriero，《神经决策树》，*arXiv*预印本 *arXiv:1702.07360*，2017年。[7] Heum Park，《使用基尼指数算法提取主要特征：来自 KAMIR 的主要不良心脏事件》，国际信息研究所（东京）·信息，第21卷，第2期，第631–638页，2018年。[8]《常数 Q 变换》，https://en.wikipedia.org/wiki/Constant-Q_transform，2019, [在线；访问日期：5月 2, 2022,]日。[9] Khaled Koutini、Jan Schlüter、Hamid Eghbal-zadeh 和 Gerhard Widmer，《使用 Patchout 高效训练音频 Transformer》，*arXiv*预印本 *arXiv:2110.05069*，2021年。[10] Jesse Engel、Cinjon Resnick、Adam Roberts、Sander Dieleman、Mohammad Norouzi、Douglas Eck 和 Karen Simonyan，《使用 WaveNet 自编码器合成音乐音符的神经音频》，载于第*34*届国际机器学习会议论文集，Doina Precup 和 Yee Whye Teh 编，国际会议中心，澳大利亚悉尼，2017年8月6–11日，第70卷，机器学习研究论文集，第1068–1077页，PMLR。[11] Mi Tian、Ajay Srinivasamurthy、Mark Sandler 和 Xavier Serra，《京剧打击乐器数据集》，2014年3月。[12] Houwei Cao、David G Cooper、Michael K Keutmann、Ruben C Gur、Ani Nenkova 和 Ragini Verma，《CREMA-D：众包情感多模态演员数据集》，*IEEE*《情感计算汇刊》，第5卷，第4期，第377–390页，2014年。[13] Eduardo Fonseca、Xavier Favory、Jordi Pons、Frederic Font 和 Xavier Serra，《FSD50K：人工标注声音事件的开放数据集》，*IEEE/ACM*《音频、语音与语言处理汇刊》，第30卷，第829–852页，2021年。