DASS: 蒸馏音频状态空间模型
是更强且更具持续性扩展能力的学习者

Saurabhchand Bhati[1], Yuan Gong[1], Leonid Karlinsky[2,3], Hilde Kuehne[3,4], Rogerio Feris[2,3], James Glass[1]

[1]MIT, USA, [2]IBM Research AI, USA, [3]MIT-IBM Watson AI Lab, USA, [4]University of Bonn, Germany

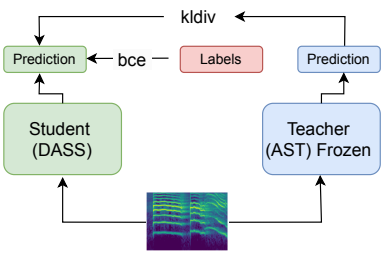
sbhati@mit.edu

ABSTRACT

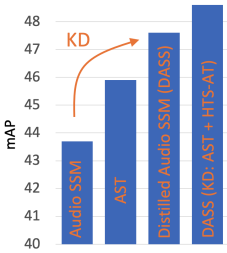
状态空间模型（SSMs）因其在长输入情况下具有高计算效率而成为音频建模中 Transformer 的替代方案。尽管近期关于音频 SSM 的工作报告了令人鼓舞的结果，但仍存在两个主要限制：首先，在 10 秒短音频标签任务中，音频 SSM 的表现仍然不如基于 Transformer 的模型如音频频谱图 Transformer（AST）。其次，虽然理论上音频 SSM 能够处理支持长音频输入，但其实际性能对长音频的评估尚未彻底进行。为了解决这些限制，在本文中，1) 我们在音频空间模型训练中应用了知识蒸馏，从而得到了一个称为 Knowledge Distilled Audio SSM（DASS）的模型。据我们所知，这是第一个在 AudioSet 上超越 Transformers 的 SSM，并达到了 48.9 的 mAP；2) 我们设计了一个新的测试称为 AudioNeedleInA H aystack (Audio NIAH)。我们发现，仅使用 10 秒音频片段训练的 DASS 可以在长达 2.5 小时的录音中检索声音事件，而 AST 模型在输入仅为 50 秒时就失败了，这表明 SSM 确实更具持续时间可扩展性。代码：GitHub，HuggingFace

1. 介绍

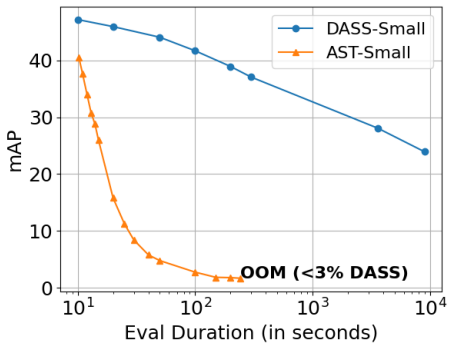
变换器已成为建模语音、文本和图像数据的主要选择 [1–5]。在过去的几年里，音频分类系统逐渐从基于 CNN 的模型 [6–9] 转向了基于变换器的模型 [10–13]。基于注意力机制的变换器模型具有更大的感受野；然而，它们具有二次计算复杂性，这使得它们不适合处理长序列 [10,11]。最近，状态空间模型（SSM）[14,15] 作为序列建模中替代基于变换器模型的选择出现，提供了一种更高效的方法，特别是对于长序列而言。与此同时，已有方法将基于 SSM 的模型适应于音频分类任务，如 AuM [16]、Audio Mamba [17] 和 SSAMBA [18]。



(a) 提出的知识蒸馏框架



(b) 音频集上的性能



(c) 音频 NIAH 性能与输入音频长度的关系

Fig. 1. (a) 我们在使用 AST 作为教师模型的音频 SSM 训练中应用了知识蒸馏。(b) 训练后的蒸馏音频 SSM（DASS）的表现优于 AST 教师模型。使用一组教师模型（AST [10] + HTS-AT [11]）进一步提升了性能。(c) DASS 可以在单个 A6000 GPU 上处理长达 2.5 小时的音频输入（> 30× 比 AST 更长），同时保持良好的性能。请注意，横轴采用对数尺度。

在本工作中，我们做出了两项主要的技术贡献：首先，现有的音频 SSM 模型 [16–18] 在音频分类任务上的性能仍然不及基于 Transformer 的模型。我们发现知识蒸馏是一种有效的增强状态空间模型的技术，使它们能够与 Transformer 模型相媲美 *超越* 。具体来说，当使用 AST 作为教师模型进行知识蒸馏训练时，我们的 DASS 模型实现了 47.6 的 mAP，优于教师模型 AST（45.9 mAP），并且模型大小减少了 1.8 倍。使用由教师模型即 AST [10] 和 HTS-AT [11] 组成的集成，DASS 模型在保持较小模型尺寸的同时实现了显著更高的 mAP，达到了 48.9。

其次，尽管最近基于 SSM 的音频模型作品 [16,18] 展示了在理论上比基于变压器的模型更节省 GPU 内存和更快推理速度的优势，但它们没有测量这些模型在较长输入长度下的性能实际。为了填补这一空白，我们设计了一个新的基准测试称为 Audio Needle In A Haystack（Audio NIAH）来衡量仅用短音频片段训练的音频分类模型在长音频上的推理性能。具体来说，我们将一根针，即一段 10 秒的音频事件，插入到更大的干草堆中，并评估模型如何准确地分类这段音频事件。在我们的实验中，我们惊讶地发现 DASS 比 AST 在长音频推理上表现得更强。具体而言，当两个模型都仅用 10 秒的音频训练时，AST 模型的表现下降至输入为 50 秒时不到 5 mAP，这是对于 10 秒输入性能的 <12%，而 DASS 的表现是 45.5 mAP（96%）在相同设置下。在一个 A6000 GPU 上，DASS 可以处理长达 2.5 小时的音频输入，并且与 10 秒输入相比仍能保持 62%的性能。

2. 相关工作

初始的音频事件分类方法依赖于 CNN [6, 7] 或者 CNN-变压器混合模型 [8, 9] 。音频频谱变换器（AST）[10] 提出了一个不使用卷积、纯注意力机制的模型，在音频事件分类任务上超越了现有的模型。层次标记语义音频变换器（HTS-AT）[11] 提出了基于 Swin 变换器的模型，减少了模型大小。Patchout faSt 频谱变换器（PaSST）[12] 提出了一种 patchout 方法来减少输入到变压器模型序列的长度。这两个模型在性能和计算需求方面都超越了 AST。音频掩码自编码器（Audio-MAE）[13] 提出了一个基于 MAE 的自我监督学习框架，并且在广泛的各类数据集上实现了最先进的音频分类性能。

对于变压器模型，计算和内存复杂度随着输入长度的增加而呈二次增长。SSMs 提供了一种替代解决方案，其复杂度与输入长度成线性关系。Gu 等人 [14] 展示了 SSMs 在建模长输入序列方面的潜力。Gu 等人 [15] 提出了一种数据依赖的选择扫描状态空间块，并扩展了 SSM 框架。他们的模型在大规模自然语言数据上优于 Transformer 模型，从而推动了 SSM 模型作为通用序列建模骨干的兴起 [19–21]。Zhu 等人 [20] 提出了一种双向 SSM，该方法结合了前向和后向 SSM 来提高对视觉数据的建模能力。Liu 等人 [21] 进一步扩展了状态空间模型，并为视觉输入提出了 2D-Selective-Scan 算法（SS2D）。SS2D 在四个方向上扫描输入：从左到右、从右到左、从上到下和从下到上。每个序列都通过选择性扫描状态空间块进行处理，输出合并以创建最终输出。

同时，一些方法将 SSM 适应于音频事件分类：AuM [16]，Audio Mamba [17] 和 SSAMBA [18]。AuM 和 Audio Mamba 将 Vision Mamba 和 VMamba 应用于音频，并展现出显著的性能和计算效率。SSAMBA 探索了用于音频的自监督 SSM 学习。与我们的工作最接近的是 Audio Mamba [17]，主要区别在于 1) 我们使用知识蒸馏并在音频事件分类上实现了最先进的性能，2) 我们提出了 NIAH 任务并评估了 SSM 时长扩展性。

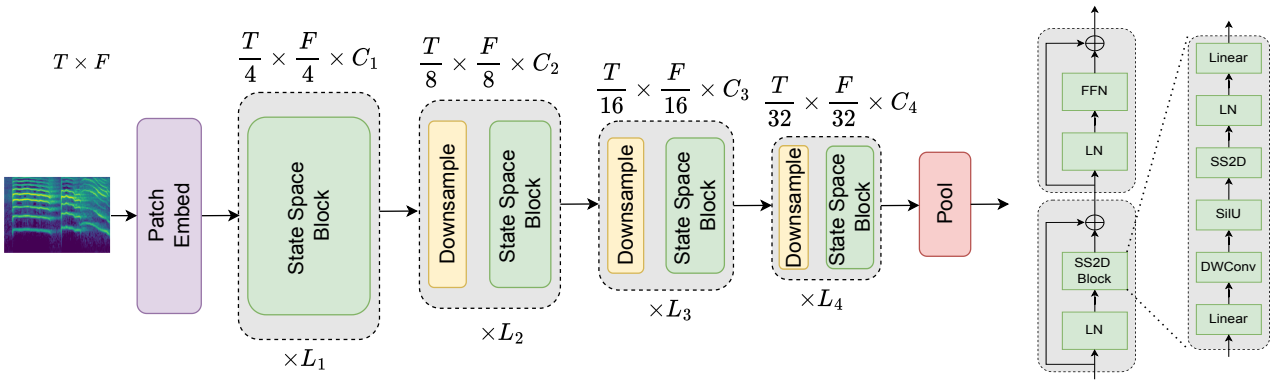
3. 蒸馏音频状态空间模型

3.1. 状态空间模型

结构化状态空间序列模型（S4）[14] 受到经典状态空间模型如卡尔曼滤波器的启发，并且与循环神经网络（RNNs）和卷积神经网络（CNNs）有广泛联系。连续状态空间模型通过线性常微分方程将一个一维函数或序列 $x(t) \in \mathbb{R} \rightarrow y(t) \in \mathbb{R}$ 映射到隐藏状态 $h(t) \in \mathbb{R}^N$，如下所示：

$$\boldsymbol{h'(t)} = \boldsymbol{A}\boldsymbol{h(t)} + \boldsymbol{B}x(t), \tag{1}$$

$$y(t) = \boldsymbol{C}\boldsymbol{h(t)} \tag{2}$$



(a) 状态空间模型在 DASS 中的架构　　(b) 状态空间块

Fig. 2. 状态空间架构在 DASS 模型中。状态空间块类似于变压器块。下采样模块减少空间维度并增加通道数量。FFN：前馈网络；LN：层归一化；SS2D：空间挤压和维数；DWConv：深度可分离卷积

其中 $\boldsymbol{A} \in \mathbb{R}^{N\times N}$ 称为演化参数，而 $\boldsymbol{B} \in \mathbb{R}^{N\times 1}, \boldsymbol{C} \in \mathbb{R}^{1\times N}$ 称为投影参数。

为了将状态空间模型适应于神经网络模型，应用了离散化方法。常用的离散化方法是零阶保持 (ZOH)，它使用时间尺度参数 Δ 将连续参数 $\boldsymbol{A}$,$\boldsymbol{B}$ 转换为离散参数 $\overline{\boldsymbol{A}}, \overline{\boldsymbol{B}}$，如下所示：

$$\bar{\boldsymbol{A}} = \exp(\Delta \boldsymbol{A}), \tag{3}$$

$$\bar{\boldsymbol{B}} = (\Delta \boldsymbol{B})^{-1}(\exp(\Delta \boldsymbol{A}) - \boldsymbol{I})\Delta \boldsymbol{B} \tag{4}$$

离散化后，状态空间方程可以重写为：

$$h_t = \overline{\boldsymbol{A}} h_{t-1} + \overline{\boldsymbol{B}} x_t \tag{5}$$

$$y_t = \boldsymbol{C} h_t \tag{6}$$

当前隐藏状态的输出仅依赖于前一隐藏状态。这种状态空间模型的观点可以被认为是与 RNNs 类比的。由于 SSM 参数不随时间变化，因此也可以将 SSM 视为一种卷积 $y_t = (x_0, x_1, ..., x_t) * (\boldsymbol{C}\overline{\boldsymbol{B}}, \boldsymbol{C}\overline{\boldsymbol{A}\boldsymbol{B}}, ..., \boldsymbol{C}\overline{\boldsymbol{A}}^{M-1}\overline{\boldsymbol{B}})$ 相同尺度的输出。下一组将特征下采样至 $\frac{T}{8} \times \frac{F}{8} \times C_2$ $\boldsymbol{x} * \overline{\boldsymbol{K}}$ 其中 $\overline{\boldsymbol{K}}$ 被称为全局卷积核。

SSMs 的一个主要优势是我们可以根据任务将它们视为 CNN 或 RNN。在推理过程中，S4 可以被视为 RNN，从而实现更快的推理和无界上下文。在训练过程中，使用卷积视图以实现类似 CNN 的并行训练。

然而，这些模型的线性时不变特性无法很好地捕获上下文信息，并且在基于内容的推理任务上表现不佳。为了解决 SSMs 的局限性，Gu 等人 [15] 提出了一种参数化方法，使时间尺度参数依赖于输入，并提出了选择性扫描 S4。然而，现在状态参数依赖于输入，无法使用卷积视图。这给高效计算带来了挑战。仍然可以推导出递归视图，并采用硬件感知并行算法来高效地计算输出。

3.2. 多维焦虑量表

我们的 DASS 模型概述如图 1 所示，图 2 显示了状态空间学生模型的详细架构。该模型可以分为四个组，每组由一个状态空间块组成，除了第一组外，所有组还有一个基于补丁合并的下采样层。该模型逐步减少空间维度并增加特征数量。一种池化方法生成频谱图的最终嵌入，然后将其传递给分类器以生成模型的最终输出。

具体而言，模型以二维频谱图 $X \in \mathbb{R}^{T\times F}$ 作为输入，然后一个补丁嵌入层提取具有空间维度 $\frac{T}{4} \times \frac{F}{4} \times C_1$ 的二维特征补丁。第一组处理该尺度上的特征并生成相同尺度的输出。下一组将特征下采样至 $\frac{T}{8} \times \frac{F}{8} \times C_2$ 维。这一过程持续两个模块，最终我们获得具有空间维度 $\frac{T}{32} \times \frac{F}{32} \times C_4$ 的特征。我们使用池化方法将信息汇总成单一嵌入，并将其传递给线性层以生成最终输出。

我们希望利用现有的基于变换器的模型来训练并提升 SSM 的性能。我们使用来自基于变换器的教师模型（AST）的知识蒸馏将知识提炼到一个基于 SSM 的学生模型（DASS）中。我们将相同的输入频谱图传递给学生（DASS）和教师（AST），并从两个模型生成输出。学生模型被训练以模仿教师模型的输出，并预测

真实标签。模型的整体损失是 $\mathcal{L} = 0.5(\mathcal{L}_{bce}(y, \hat{y}_{stu}) + \mathcal{L}_{kldiv}(\hat{y}_{teach}, \hat{y}_{stu}))$，其中 $\mathcal{L}_{bce}$ 是学生 DASS 模型输出 $\hat{y}_{stu}$ 与真实标签 y 之间的二元交叉熵损失，而 $\mathcal{L}_{kldiv}$ 是从 $\hat{y}_{stu}$ 到教师模型输出 $\hat{y}_{teach}$ 的 KL 散度。

4. 实验

我们评估了 DASS 模型在弱标记音频事件分类任务上的性能。我们使用 AudioSet 数据集来训练和评估我们的模型。数据集和训练细节将在以下章节中描述。

4.1. 数据集和训练详情

AudioSet [22] 包含从 YouTube 视频中提取的超过 200 万个 10 秒音频片段。这些声音片段被标记为一组包含 527 个标签中的一个。完整的训练数据集（AS-2M）、平衡数据集（AS-20K）以及评估数据集分别包含 200 万、2 万和 2.2 万个数据点。我们遵循 AST [10] 的训练流程，但使用了不同的学习率。对于平衡数据集和完整数据集，我们都采用了 1e-4 的学习率。对于平衡数据集，我们进行了 25 轮的训练，并且从第 10 轮开始每隔 5 轮将学习率减半。对于完整的训练数据集，我们在前两个轮次之后每一轮都将学习率减半，总共进行 10 轮训练。我们使用 Adam 优化器和大小为 12 的批量来训练模型。

我们对两种不同的模型进行了实验：DASS-Small 和 DASS-Medium，它们分别在四个组中包含 (2, 2, 8, 2) 和 (2, 2, 15, 2) 层。两个模型的特征维度均为 C_1, C_2, C_3, C_4= (96,192,384,768)，分布在四个组中。DASS-Small 和 DASS-Medium 分别包含 3000 万和 4900 万个参数。

4.2. 预训练和知识蒸馏的影响

我们将 ImageNet 预训练的 DASS 与随机初始化的 DASS 进行了比较，有无知识蒸馏的情况均有涉及。如表 1 所示，ImageNet 预训练模型的表现优于随机初始化的 DASS 模型。

知识蒸馏在两种设置中都提高了性能：当我们使用预先训练的图像数据模型时，以及我们不使用时。没有访问 ImageNet 预训练模型的情况下，性能提升更高。我们探索了两种不同的知识蒸馏损失函数：KL 散度和二元交叉熵。它们表现相似，因此我们在所有实验中都使用 KL 散度。为了进一步探索知识蒸馏的好处，我们继续对一个随机初始化的 DASS 模型进行了 100 个周期的训练。该模型在 Audioset 平衡集上实现了 31.6 的 mAP，与带有预训练权重的 AST 相同，并且接近于使用 Imagenet 预训练但没有知识蒸馏的 DASS 模型的表现。通过知识蒸馏训练的 DASS 模型甚至超过了参数量明显多于 DASS 模型的基础 AST 模型。

	IN Pretrain	KD	mAP
AST-Small (23M)	False	False	10.6
AST-Base (86M)	False	False	14.8
DASS-Small (30M)	False	False	12.0
DASS-Small (30M)	False	True	20.2
AST-Small (23M)	True	False	31.0
AST-Base (86M)	True	False	34.7
DASS-Small (30M)	True	False	34.6
DASS-Small (30M)	True	True	38.4

Table 1. DASS 与 AST 在 AS-20K 上的比较。知识蒸馏（KD）在有无预训练的情况下都是有益的。

4.3. 更强的老师是否造就更好的学生

为了训练 DASS 模型，我们使用 AST 基础模型作为教师。DASS 模型的表现优于 AST 教师模型。在本节中，我们探讨是否使用更强的 DASS 教师会产生更强的学生。为了减少计算成本，我们将训练限制为 5 个周期。如表 2 所示，使用更强大的教师并不会导致学生表现更好，尽管可能学生从具有不同架构的教师那里获益更多。先前的研究观察到基于 CNN/Transformer 的师生模型存在类似的趋势 [23]。

4.4. AudioSet：与其他方法的比较

我们比较了所提出的 DASS 模型与基于变压器的最先进方法和同时期的 SSM 基础方法。如表 3 所示，DASS 优于所有现有方法。DASS 的参数少于 AST 和 audio-MAE（86M 对 49M），并且训练所需的计算资

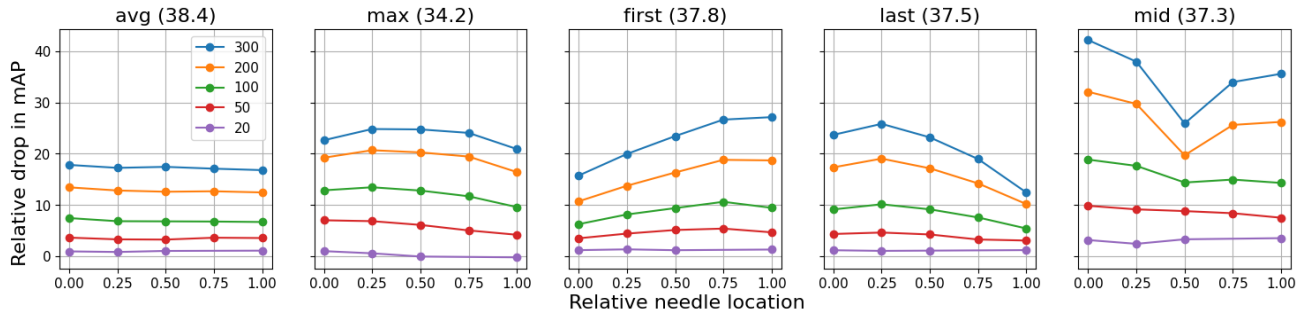


Fig. 3. DASS-Small 在 AS-20K 上训练后在 NIAH 任务中的池化策略性能。括号显示了前 10 名的性能。

model		mAP	
Student	Teacher	Teacher	Student
DASS-Small	AST-Base	45.9	47.1
DASS-Medium	AST-Base	45.9	47.1
DASS-Small	DASS-Small	47.1	46.9
DASS-Small	DASS-Medium	47.1	46.7
DASS-Medium	DASS-Small	47.1	47.1
DASS-Medium	DASS-Medium	47.1	46.9

Table 2. 不同教师在 AS-2M 上的 DASS 学生模型性能比较

源显著较少（audio-MAE 需要 64 个 V100，而 DASS 只需要一个 A5000）。DASS 和其他基于 SSM 的模型在推理时也更快，并且比基于变压器的模型所需的计算资源更少。

通过使用知识蒸馏来训练 SSMs，我们弥合了变压器和 SSMs 之间的性能差距。DASS 提供了两全其美的效果：它超越了最先进的基于变压器的模型，同时具有处理长音频序列的效率和能力。

5. 音频针在干草堆中：在较长序列上的评估

SSMs 的主要优势之一是它们能够以线性复杂度处理较长的序列 [15] ，这与 Transformers 不同。最近基于状态空间的模型在音频事件分类任务上表现出色。其中一些模拟了更长的输入序列来比较速度和 GPU 内存使用情况，与基于 Transformer 的模型如 AST [13,16] 进行对比。尽管他们的简要分析表明，状

	Params	Pretrain	mAP
Transformer based models			
AST [10]	87M	IN SL	45.9
HTS-AT [11]	31M	IN SL	47.1
PaSST [12]		IN SL	47.1
Audio-MAE† [13]	86M	SSL	47.3
BEATs(iter3) [24]	90M	SSL	48.6
EAT [25]	88M	SSL	48.6
Concurrent SSM models			
AuM [16]	26M	IN SL	39.7
Audio Mamba [17]	40M	IN SL	44.0
DASS-Small	30M	IN SL	47.2
DASS-Medium	49M	IN SL	47.6
Teacher ensemble: AST + HTS-AT			
DASS-Small	30M	IN SL	48.6
DASS-Medium	49M	IN SL	48.9

Table 3. 性能比较于 AS-2M。在 SL 中：ImageNet 监督学习；SSL：自监督学习；† 工业级计算

态空间模型理论上可以处理比 AST 更长的输入序列，但他们并没有测量这些长度下状态空间模型的表现。

我们 SSM 研究的一个目标是实验量化 DASS 模型的时长可扩展性。为此，我们仅在 10 秒频谱图上训练 DASS。在评估期间，我们合成更长的输入序列以测量模型的时长稳健性。

我们设计了一个音频针在草堆中（Audio Needle in A Haystack，简称 Audio NIAH）任务，在该任务中我们将一根针：一段来自 Audioset 数据集的 10 秒音频频谱图随机插入到不同长度的大草堆中。我们通

过两种方式构建草堆：首先通过零填充来达到所需的长度，其次在不同的级别上附加噪声。对于后者，我们在各种信噪比（SNR）下生成噪声波形，然后使用与生成针的滤波器组（fbank）特征相同的流程生成滤波器组特征。在这两种情况下，针（10 秒音频频谱图）都不会被修改。如前所述，模型仅在 10 秒的频谱图上进行训练，在评估期间执行 NIAH 测试以测量模型的时间长度鲁棒性。

5.1. NIAH：池化方法的影响

我们的第一次 NIAH 实验探讨了针的位置是否会导致性能差异，或者 DASS 模型能否从草堆中的任何位置获取信息。为此，我们创建了一个不同长度的草堆，并在不同的相对位置插入 10 秒频谱图。我们在草堆中将针插入以下位置：0（开始处）、0.25、0.5（中间）、0.75 和 1.0（结束处）。我们通过测量相对于 DASS 模型基线性能的相对下降来衡量针的位置的影响，即当不使用草堆而只是将 10 秒频谱图作为输入传递给 DASS 模型时：$(mAP_{10} - mAP_t)/(mAP_{10})$。

AST 模型依赖一个 CLS 标记来总结信息，而 SSMs 提供了一种自然的方式将信息汇总到单一的嵌入中。例如，最后一个嵌入概括了迄今为止看到的所有信息。双向特性允许我们在任何时间步使用嵌入作为 CLS 嵌入。

DASS 使用池化方法生成 CLS 标记并在分类器之前汇总信息。我们探讨了池化策略对 NIAH 任务的影响。如图 3 所示，池化方法的选择会根据相对针位置影响性能。

当模型使用第一个标记嵌入作为 CLS 标记进行训练时，如果针位于开头，它在 NIAH 任务上表现更好。类似地，分别以最后和中间位置作为 CLS 标记训练的 DASS 模型，在针的位置为最后或中间时性能下降幅度最小。平均池化对针位置的相对位置表现出最低的敏感度。我们还探索了将求和作为池化机制，但这导致性能严重下降，尤其是在较长的时间段内。

对于所有池化方法，性能下降随着草堆长度的增加而增加。这一结果是有道理的，因为模型必须处理和总结越来越长的话语。尽管如此，对于 DASS 模型，在 50 秒的草堆长度下，性能下降不足 5%，而 AST 在这些长度上几乎失去了大部分性能。

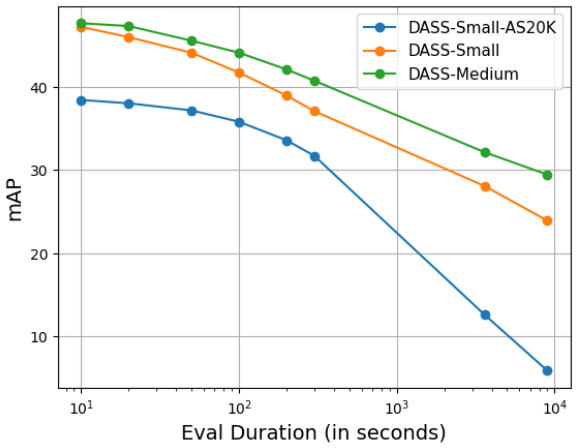


Fig. 4. mAP 性能在不同长度的麦垛上的表现。麦垛：零填充，针的位置：0.5

5.2. NIAH：草堆长度对性能的影响

我们的第二个 NIAH 实验探讨了 DASS 模型的极限。在这个实验中，我们将针的位置固定在草堆的中间，并将草堆长度从 10 秒增加到长达 2.5 小时（评估时间相比于训练时间增加了 900 倍）。我们在单个 A6000 GPU 上进行了这个 2.5 小时的实验。

如图 4 所示，更强的模型对长度变化显示出更高的鲁棒性。在较小平衡子集上训练的小型模型与在完整 AudioSet 上训练的小型模型相比性能显著下降。在完整 AudioSet 上训练的中型模型，其表现类似于小型模型，甚至对评估长度的变化表现出更高的鲁棒性。过度参数化有助于 DASS 模型更好地适应时长变化。在极端长度（即 2.5 小时）下，中型模型的表现与在平衡子集上训练的小型模型相似。

我们在基于变换器的 AST 模型上进行了一项类似的实验。为了避免训练和测试期间学习到的位置嵌入之间的不匹配，我们使用正弦嵌入训练了一个 AST 模型。这使我们能够将输入和位置嵌入扩展到任意长度。对于 AST，我们将针只放在 haystack 的开头，因此在训练和测试过程中添加到针的位置嵌入之间没有不匹配。

AST 模型的性能随着基准文本长度的增加而迅速下降。在仅 30 秒的评估长度下，其性能降至 10 秒长度时性能的 20%，而这种情况在 DASS 模型中即使

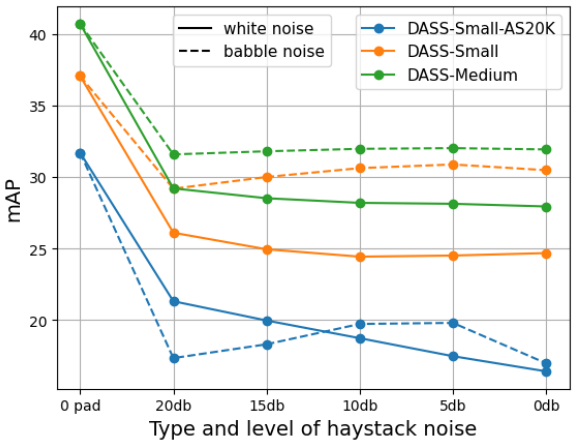


Fig. 5. mAP 对比不同类型和级别的白噪声或粉红噪声在信号中的表现。信号长度：300 秒

在长达 2.5 小时的评估长度下也不会发生。我们相信这是由于注意力权重被分散到更多标记上从而变得非常小所致。未来，我们希望分析 AST 在各种评估持续时间下的注意力图。

这些结果特别有用，因为我们在较长的输入序列上不需要训练 DASS 模型，从而减少了训练和使用这些模型处理诸如视频等较长数据模式所需的总体计算资源。

5.3. NIAH：草堆构建的影响

如前所述，我们通过两种方式构建稻草堆：零填充和噪声填充。这里我们探讨 DASS 模型在两种不同稻草堆构造下的表现。我们研究了两种类型的噪声：高斯噪声和混响噪声。混响噪声是通过结合 MUAVIC 数据集 [26] 中的 30 个音频混响噪声文件构建的。

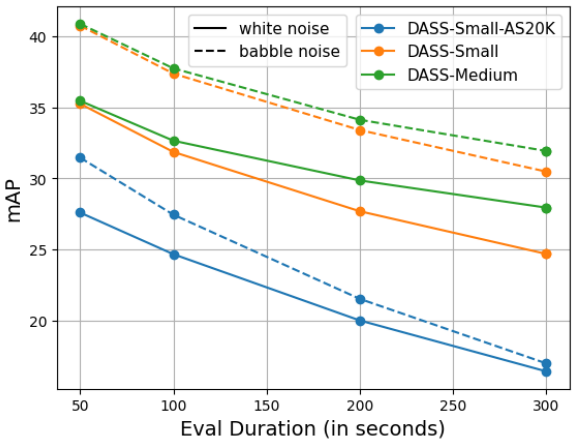


Fig. 6. mAP 性能在不同长度的干草堆上的表现。干草堆：信噪比为 0 的白噪声，针的位置：0.5。

对于两种噪声，DASS 模型的表现都比零填充的草堆差。对于混响噪声，我们看到混合结果，当信噪比下降时性能上升，然后在 0 信噪比时再次下降。对于白噪声，从 20 信噪比降到 15 时性能下降，之后则保持相对稳定。

我们相信这一结果是因为 AudioSet 数据集将语音和白噪声作为其中的一个类别，而 DASS 模型可能正确地将背景噪音分类为语音或噪声，这影响了性能。我们观察到，在混响噪声下的表现有更大的波动，因为 AudioSet 中标记为语音的数据点显著多于标记为白噪声的数据点。对于两种噪声设置，更强的模型表现出对噪声更具鲁棒性。

为了测量时长对基于噪声的稻草堆的影响，我们将针的位置固定在中间，将噪声信噪比固定为 0，并改变稻草堆的长度。如图 6 所示，性能随着长度的增加而下降。对于基于噪声的稻草堆而言，性能下降明显大于零填充的稻草堆。300 秒时带有基于噪声的稻草堆的 DASS 模型的表现不如使用零填充的 2.5 小时稻草堆。

6. 结论

在本文中，我们提出了 DASS：一种基于状态空间的模型，实现了音频事件分类领域的最新结果。DASS 具有计算密集型变压器模型的性能和状态空间方法的效率。我们的实验表明，知识蒸馏对于不同数据集大小和模型规模都有帮助。DASS 在表现上超过了基于 AST 的教师模型。

音频 SSM 理论上可以支持更长的语音输入。我们提出了一项音频 NIAH 任务，以衡量训练和评估长度之间时长不匹配对性能的影响。DASS 在训练和评估过程中比 AST 更能显著地抵御时长不匹配的影响。我们还观察到，更强的 DASS 模型往往更能抵抗时长不匹配的影响。

7. REFERENCES

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.

[2] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever, "Robust speech recognition via large-scale weak supervision," in International conference on machine learning. PMLR, 2023, pp. 28492–28518.

[3] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," arXiv preprint arXiv:2010.11929, 2020.

[4] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou, "Training data-efficient image transformers & distillation through attention," in International conference on machine learning. PMLR, 2021, pp. 10347–10357.

[5] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al., "Llama: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, 2023.

[6] Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2880–2894, 2020.

[7] Yuan Gong, Yu-An Chung, and James Glass, "Psla: Improving audio tagging with pretraining, sampling, labeling, and aggregation," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 29, pp. 3292–3306, 2021.

[8] Koichi Miyazaki, Tatsuya Komatsu, Tomoki Hayashi, Shinji Watanabe, Tomoki Toda, and Kazuya Takeda, "Convolution-augmented transformer for semi-supervised sound event detection," in Proc. workshop detection classification Acoust. Scenes events (DCASE), 2020, pp. 100–104.

[9] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark D Plumbley, "Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2450–2460, 2020.

[10] Yuan Gong, Yu-An Chung, and James Glass, "Ast: Audio spectrogram transformer," arXiv preprint arXiv:2104.01778, 2021.

[11] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov, "Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection," in ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022, pp. 646–650.

[12] Khaled Koutini, Jan Schlüter, Hamid Eghbal-Zadeh, and Gerhard Widmer, "Efficient training of audio transformers with patchout," arXiv preprint arXiv:2110.05069, 2021.

[13] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian

Metze, and Christoph Feichtenhofer, "Masked autoencoders that listen," Advances in Neural Information Processing Systems, vol. 35, pp. 28708–28720, 2022.

[14] Albert Gu, Karan Goel, and Christopher Ré, "Efficiently modeling long sequences with structured state spaces," arXiv preprint arXiv:2111.00396, 2021.

[15] Albert Gu and Tri Dao, "Mamba: Linear-time sequence modeling with selective state spaces," arXiv preprint arXiv:2312.00752, 2023.

[16] Mehmet Hamza Erol, Arda Senocak, Jiu Feng, and Joon Son Chung, "Audio mamba: Bidirectional state space model for audio representation learning," arXiv e-prints, pp. arXiv–2406, 2024.

[17] Jiaju Lin and Haoxuan Hu, "Audio mamba: Pretrained audio state space model for audio tagging," arXiv preprint arXiv:2405.13636, 2024.

[18] Siavash Shams, Sukru Samet Dindar, Xilin Jiang, and Nima Mesgarani, "Ssamba: Self-supervised audio representation learning with mamba state space model," arXiv preprint arXiv:2405.11831, 2024.

[19] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur, "Long range language modeling via gated state spaces," arXiv preprint arXiv:2206.13947, 2022.

[20] Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang, "Vision mamba: Efficient visual representation learning with bidirectional state space model," arXiv preprint arXiv:2401.09417, 2024.

[21] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu, "Vmamba: Visual state space model," arXiv preprint arXiv:2401.10166, 2024.

[22] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2017, pp. 776–780.

[23] Yuan Gong, Sameer Khurana, Andrew Rouditchenko, and James Glass, "Cmkd: Cnn/transformer-based cross-model knowledge distillation for audio classification," arXiv preprint arXiv:2203.06760, 2022.

[24] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei, "Beats: Audio pre-training with acoustic tokenizers," arXiv preprint arXiv:2212.09058, 2022.

[25] Wenxi Chen, Yuzhe Liang, Ziyang Ma, Zhisheng Zheng, and Xie Chen, "Eat: Self-supervised pre-training with efficient audio transformer," arXiv preprint arXiv:2401.03497, 2024.

[26] Mohamed Anwar, Bowen Shi, Vedanuj Goswami, Wei-Ning Hsu, Juan Pino, and Changhan Wang, "Muavic: A multilingual audio-visual corpus for robust speech recognition and robust speech-to-text translation," arXiv preprint arXiv:2303.00628, 2023.