SED-RECIPES: UNDERSTANDING THE IMPACT OF MASKING STRATEGIES, SCHEDULES AND DECODERS ON SOUND EVENT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent work has shown that sound event detection (SED) benefits from in-domain, self-supervised pretraining of the decoder. Different works propose different masking strategies, decoders and curricula, but no work has systematically compared the different recipes. In this work, we conduct one of the largest systematic evaluations of different configurations. In addition, we introduce three novel masking strategies that are inspired from the inductive biases of SED and new evaluation metrics that elucidate the impact of polyphonicity and event duration. Our findings show that shorter events remain challenging and that saliency-guided masks can improve performance compared to standard, randomized masking.

1 INTRODUCTION

Sound event detection (SED) is a foundational skill for machines that understand sound Mesaros et al. (2021). It entails the identification and categorization of individual sound events irrespective of their own duration and the amount of co-occurring sound sources (*polyphonicity*). As with many other fields of artificial intelligence, SED has benefited tremendously from the proliferation of deep learning (DL) approaches Triantafyllopoulos et al. (2025). Given its sequential, many-to-many nature, it is well-suited to sequential models, with recurrent neural networks (RNNs) and transformers being the dominant modeling paradigms. However, the annotation of SED data is incredibly resource-intensive Harju et al. (2024), as *strong* labels require each frame of the audio to be annotated Martín-Morató & Mesaros (2023). Moreover, the correct identification of sound sources is partially subjective, and becomes increasingly difficult as the amount of polyphonicity increases. As a result, the size of publicly-available SED datasets is substantially smaller than that found in other domains. Consequently, the community has long adopted *unsupervised pretraining* as a mitigation strategy to increase performance Liu et al. (2022). Most recently, there has been increasing interest in self-supervised learning (SSL) to improve learnt representations. These approaches are largely inspired by advances in the fields of natural language processing and computer vision. For instance, 'BERT-style' masking and reconstruction has been successfully transferred to SED Cai et al. (2024b).

Crucially, SED has two unique characteristics which distinguish it from other domains: a) It is an inherently *temporal* task, requiring a precise identification of event boundaries, more akin to image segmentation than classification tasks. b) It is characterized by *polyphonicity*, with multiple, co-occurring sources; importantly, these sources do not completely occlude one another, but are all superimposed on the same input signal. This raises the question of how the SSL paradigm should be adapted to the domain of SED.

Inspired by this gap in previous research, our work introduces the following contributions: a) We first stratify SED performance per event duration and amount of overlapping sound events; this disaggregation allows us to quantify the performance of different SED models along those two dimensions; b) We then conduct a large-scale ablation study to investigate how common SED pipelines perform across those two dimensions; c) Moreover, we introduce a set of domain-inspired SSL masking strategies aimed to improving performance for shorter time events and polyphonic sequences; d) Finally, we open-source our code which streamlines the execution over different SED recipes comprising different masking strategies, curricula and decoders. The remainder of our work is structured as follows: We start by introducing related work in audio SSL, and the modern recipe

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

design of sound event detection. After introducing the training regime, domain, and tasks involved in the method section, we discuss the set of masking strategies that are considered in this work, and contrast experimental results with related work. We conclude by providing a summary and outlook for masked reconstruction representation learning for frame-wise tasks.

2 Related Work

SSL aims to learn useful representations by leveraging properties of the input signal to create prediction targets. The most common SSL tasks rely either on cross-modal information De Sa (1993) or on the *locality bias*: for images, pixels that are close together, often belong to the same object Marr (1982), whereas for language, neighboring words Mikolov et al. (2013), or neighboring sentences Devlin et al. (2019), can be assumed to be related. These inductive biases have powered the tremendous success of SSL approaches for computer vision and natural language processing.

The locality assumption does not always hold for auditory streams. Different sound sources can be simultaneously active and partially overlap in frequency. As such, common masking strategies fall short of the objective to disentangle different sound events, a necessary prerequisite for improving polyphonic SED performance. In addition, shorter sound events suffer from the same representational issues plaguing smaller objects in image segmentation Oquab et al. (2024) - masking approaches that mask out large regions of an audio signal may completely remove a short audio event, causing pretrained models to disregard them in their representations.

Nevertheless, there has been a plethora of SSL approaches for general audio tasks, or even tailored for the SED domain. In general, we identify three key distinguishing elements for modern SED recipes:

Encoders: Trainable encoders are responsible for extracting generalizable representations. Early works primarily relied on CNN-based encoders, which were later substituted by transformer-based encoders and audio state-space models (SSMs). In an attempt to increase performance, most recipes use a dual-encoder approach, where embeddings are first separately extracted from a CNN and a transformer encoder and then fused to derive a joint representation.

Decoders: Extracted features are processed by temporal models to extract sound event information. Typically, RNNs (LSTMs; GRUs) are used to map features to frame-level predictions. One contribution of our work is that we conduct the largest ablation study to date where we include different decoder architectures, including transformers and SSMs and study their impact on SED performance.

Training stages: SED training is conducted in three stages:

Task-agnostic SSL pretraining: First, encoders are pretrained on large corpora (e. g., AudioSet) to learn generalizable representations. These representations are task-agnostic and are primarily geared towards *audio tagging* (the identification of sound presence without strict temporal boundaries).

SED-specific SSL pretraining: Recent works have introduced an intermediate SSL pretraining stage which further adapts these generic encoders to SED without labels. **The major contribution of our work is that we systematically study the impact of this second SSL stage while additionally introducing new SSL strategies that are inspired by the inductive biases of polyphonic SED.**

SED finetuning: Finally, learned representations are connected to a decoder and the whole system is trained end-to-end with a supervised objective. Oftentimes, unlabeled data is also leveraged in this stage in *semi-supervised learning* fashion, where pseudo-labels are derived by a teacher model (typically, an exponential-average of student model weights) and used to guide the current model. We also evaluate the impact of this stage on multiple architectures.

JiTTER and PMAM are close related, recent studies for SED-specific SSL systems based on MAT-SED. JiTTER replaces masked prediction with a hierarchical temporal shuffle. On DESED it reports a 5.89% PSDS gain over MAT-SED under that protocol Nam & Park (2025). PMAM supervises a masked model with prototype pseudo-labels and reports $\text{PSDS}_1{=}.625$ on its own DESED setup Cai et al. (2024a). These works use different configs and environments compared to our fixed PaSST-CNN ablation in this paper, but also build on top of the MAT-SED framework Cai et al. (2024b).

3 METHOD

3.1 TASK, DATA, AND PIPELINE

We focus on DESED Turpault et al. (2019) as the standard polyphonic SED benchmark (DCASE Task 4). The corpus covers 10 domestic event classes and keeps the original strong, weak, synthetic, and in-domain unlabeled splits: 3373 strongly labeled real clips, 1578 weakly labeled clips, 10,000 synthetic strongly labeled clips, and 14,412 unlabeled in-domain clips. Unless noted, downstream metrics are reported on the 1168-clip real validation set with the teacher model. Log-mel inputs are extracted at 32 kHz with 128 bins on fixed 10 s clips. We use a 25 ms window (800 samples), 10 ms hop (320 samples), and FFT size 1024, yielding T=1000 frames per clip.

Three-stage training. Training follows the MAT-SED recipe Cai et al. (2024b). **Stage 1 (PT1)** performs masked latent reconstruction on DESED with a frozen PaSST encoder and trainable temporal decoder. PT1 batches mix strong real, weak, and in-domain unlabeled clips. **Stage 2 (FT1)** adds the CNN branch and trains the CNN and classification head while keeping PaSST and the PT1 decoder frozen. **Stage 3 (FT2)** finetunes the full hybrid model on strong, weak, and unlabeled data, with mean-teacher consistency Tarvainen & Valpola (2018) on the weak and unlabeled streams. The teacher is an EMA of the student (β=0.999) and uses sliding-window encoder views in FT2.

Architecture by stage. PT1 uses a PaSST-only encoder initialized from AudioSet-pretrained weights. PaSST layer-10 patch tokens are mean-pooled over frequency and interpolated to T=1000 frame tokens before a temporal decoder. For downstream training, we use a hybrid PaSST-CNN architecture and the PT1/FT1/FT2 schedule described below. The PaSST branch follows the same layer-10 pooling path as described in MAT-SED Cai et al. (2024b). Additionally, we use a parallel 8-block CNN mel encoder that extracts local frame features. Both streams are linearly projected to a shared width and fused as $x = x_{\mathrm{PaSST}} + \alpha\, x_{\mathrm{CNN}}$ with fixed α=0.5. The fused representation is passed to the same temporal decoder with clip-level weak and frame-level strong heads. Unless noted, the decoder is a three-layer Transformer-XL with relative positional encoding.

Augmentation and optimization. PT1 applies FilterAugment-style band gains. FT1 and FT2 additionally use SpecAugment Park et al. (2019) and mel-view mixing with rate 0.5. We optimize with the AdamW optimizer and an ExponentialDown schedule. PT1 trains the decoder at 10^{-4} for 20 epochs. FT1 uses learning rates of 10^{-4} on the CNN and 2×10^{-4} on the head, for 15 epochs. FT2 uses 10^{-4} on the CNN, decoder, and head, and 5×10^{-6} on PaSST, for 20 epochs.

3.2 MASKED LATENT MODELING

Our PT1 objective follows MAT-SED Cai et al. (2024b) latent MLM between encoder and temporal decoder. Like SSAST Gong et al. (2022), the encoder always receives the full mel spectrogram. Unlike MAE-AST Baade et al. (2022), masking is not applied at the encoder input. Corruption instead targets 1D pooled frame tokens $z_t \in \mathbb{R}^C$ between encoder and decoder. Corrupted tokens follow 80/10/10 mask-random-keep replacement Devlin et al. (2019). In the experiments, the default mask rate is r=0.75. Some strategies have a dynamic rate, resulting in $r \leq 0.75$. The PT1 loss is the latent mean squared error (MSE) computed on masked positions only:

$$\mathcal{L}_{\mathrm{MSM}} = \frac{1}{|\mathcal{M}|} \sum_{t \in \mathcal{M}} \|\widehat{\mathbf{z}}_t - \mathbf{z}_t\|_2^2. \tag{1}$$

Decoders. Most of our work uses Transformer-XL Dai et al. (2019) as the decoder, following MAT-SED, where it was the strongest among the attention-based temporal heads they evaluated Cai et al. (2024b). However, we additionally benchmarked different transformer and RNN-based backbones for comparison, such as Conformer Gulati et al. (2020) and RNN-based heads. Importantly, we also evaluated standard State-Space Models (SSMs) and xLSTMs as decoders in a full SED pipeline, namely, Mamba Gu & Dao (2024), BiMamba Gao & Yin (2025), and Hydra Hwang et al. (2024).

3.3 EVALUATION

We report PSDS_1 and PSDS_2 Bilen et al. (2020) with per-class median filtering before export, plus event-based F1 where noted Mesaros et al. (2016). We also report MSE on masked latent reconstruction to quantify PT1 performance. The suffix /t in tables and figures denotes evaluation with the EMA teacher model.

4 MASKING STRATEGIES

All our masking strategies are evaluated under a masked language modeling (MLM) objective. Concretely, the PaSST encoder maps a log-mel clip to frame-wise latent embeddings $\mathbf{Z} \in \mathbb{R}^{B\times T\times C}$, where B is batch size and C is latent width. Below the batch index is suppressed, and each mask is defined for one clip. These embeddings are subsequently masked by each masking algorithm. A temporal decoder is then trained to reconstruct the ground truth embeddings at the corrupted positions. In this section, we present existing mask strategies and introduce our proposed masking schemes. We also discuss masking schedules and combinations over different masking strategies. Each strategy selects a boolean mask over *frame indices* $t \in \{0, \dots, T-1\}$ and corrupts the corresponding latent rows. Rather than zero out masked frames, we follow BERT Devlin et al. (2019): each selected frame $t \in \mathcal{M}$ is replaced by a learned mask embedding with probability 0.8, a random latent from the same batch with probability 0.1, or left unchanged with probability 0.1. The different strategies differ only in how the mask index set, $\mathcal{M}$, is drawn for each step.

Existing masking strategies can be grouped into *geometry-based*, *saliency-based* and *curriculum-based* strategies. Geometry schemes draw mask indices based on geometry alone Bao et al. (2022); Baevski et al. (2023), saliency schemes bias mask placement toward informative regions Kakogeorgiou et al. (2022), while curriculum schemes adapt mask difficulty throughout training to focus on progressively harder regions Seth et al. (2024); Wang et al. (2023). In the following, T denotes decoder frames and r=0.75 is the target mask rate. Each strategy uses T=1000 and returns boolean indicators m_t which comprise the mask index set $\mathcal{M} = \{t : m_t{=}1\}$.

Shared forward pass. Every masking strategy executes the same three steps on each training batch, acting on the encoder output:

1. Draw a boolean mask $\mathbf{m} \in \{0, 1\}^T$.
2. Form $\mathcal{M}$ and record the realized selected fraction $r_{\text{eff}} = |\mathcal{M}|/T$.
3. Apply the strategy's corruption to obtain $\widetilde{\mathbf{Z}}$; BERT replacement uses the $80/10/10$ rule above, while TEM also uses temporal permutation.
4. Compute $\widehat{\mathbf{Z}} = f_{\text{dec}}(\widetilde{\mathbf{Z}})$ and evaluate $\mathcal{L}_{\text{MSM}}$ on the selected positions.

4.1 GEOMETRY-BASED MASKING

Past research has shown that structured masking strategies, like block masking in BEiT Bao et al. (2022) and inverse block masking (IBM) in data2vec 2.0 Baevski et al. (2023), improve the quality of learned representations beyond random masks, as proposed in Masked Auto-Encoders (MAE) He et al. (2022). We introduce them as standard baselines.

Random Masking. Random masks depend on independent Bernoulli draws, $m_t \sim \text{Bernoulli}(r)$. One major downside is that they break temporal contiguity. Nevertheless, they form a natural, naive baseline. The random mask index is defined as:

$$\mathcal{M}_{\text{rand}} = \{t : m_t{=}1\}. \tag{2}$$

Block. Block masking begins by grouping frames in contiguous segments of width w. For segment starts $S_w = \{jw : j \in \mathbb{Z}_{\geq 0},\ jw < T\}$, block masking defines a gate $u_s \in \{0, 1\}$ for each $s \in S_w$. Segment gates are sampled at the segment level so that $|\mathcal{M}_{\text{block}}|/T \approx r$.

Default experiments use a block width of $w \in \{4, 16\}$. A segment is masked iff $u_s{=}1$, giving

$$m_t = u_{w\lfloor t/w \rfloor},$$
$$\mathcal{M}_{\text{block}}(w) = \bigcup_{\substack{s \in S_w \\ u_s=1}} \{t \in \{0, \ldots, T{-}1\} : s \le t < s + w\}. \tag{3}$$

Inverse block. Inspired by inverse block masking (IBM) in data2vec 2.0 Baevski et al. (2023), we use a *complement* of standard block masking rather than the original preserve-blocks sampler. In our case, *visible* denotes uncorrupted latent frames, rather than encoder input patches. Concretely, we first draw a block mask at rate r and take its complement,

$$\mathcal{M}_{\text{inv}}(w) = \{0, \ldots, T{-}1\} \setminus \mathcal{M}_{\text{block}}(w)\,. \tag{4}$$

With $r{=}0.75$, the realized masked fraction is therefore about $1{-}r{=}0.25$. This IBM-like geometry leaves long contiguous spans intact and corrupts only short gaps. The decoder must reconstruct the short gaps from surrounding uncorrupted latents, which encourages clip-level temporal reasoning, rather than local bias.

4.2 Saliency-based Masking

Saliency-based masking focuses on characteristics of the input signal. A classic example is AttMask, which places additional emphasis towards regions that receive higher attention from the encoder Kakogeorgiou et al. (2022). Motivated by this, we introduce three new SED-inspired masking strategies: temporal entropy masking (TEM), event-aware masking (EA) and energy-reweighted masking (ERB).

AttMask. We adapt AttMask Kakogeorgiou et al. (2022) to 1D frame tokens. AttMask places the mask budget on frames with high encoder attention. PaSST self-attention maps from the first L layers (default $L{=}6$) are averaged to a per-frame score a_t. A high-attention pool of size $\max(rT, \lfloor \rho\, T \rfloor)$ is taken from the largest a_t values. If that pool exceeds rT, we subsample it uniformly so $|\mathcal{M}|/T \approx r$. Placement is therefore attention-guided but *unstructured*: mask indices are independent frames, not contiguous blocks. The reported runs use $\rho \in \{0.30, 0.50\}$.

TEM (temporal entropy masking). TEM is a *label-free* masking strategy leveraging token-level PaSST activations, $\bar{z}_t$. Salience is defined as a combination of normalized activations and their temporal change:

$$s_t = \tfrac{1}{2}\,\frac{\bar{z}_t}{\max_{t'} \bar{z}_{t'}} + \tfrac{1}{2}\,\frac{|\bar{z}_t - \bar{z}_{t-1}|}{\max_{t'} |\bar{z}_{t'} - \bar{z}_{t'-1}|}. \tag{5}$$

Let $\mathcal{H}$ contain the top fraction $\rho{=}0.30$ of frames ranked by s_t. TEM applies different corruptions to high- and low-salience regions. A fraction $\tau{=}0.60$ of frames in $\mathcal{H}$ is corrupted by local temporal permutation, inspired by JiTTER frame shuffling Nam & Park (2025). We empirically chose $\rho{=}0.30$ and $\tau{=}0.60$ as the TEM operating points. On the complement, $\mathcal{H}^{\text{c}}$, we applied BERT-style masking at rate r. The intent is to stress temporal-order recovery on high-salience regions and high-fidelity reconstruction on low-salience regions.

EA (event-aware masking). EA is a partially supervised strategy that biases masks toward event-active frames using strong, weak, and unlabeled data. On strongly labeled clips, $e_t \in \{0, 1\}$ is the union of active classes. In the absence of labels, we fall back to spectral flux (average difference in energy between successive mel frames). Masks are subsequently sampled for the most informative regions, again, with the option to create block masks. The mask budget splits between event and background with a background ratio $\beta{=}0.15$,

$$k_{\text{evt}} = \text{round}\big((1{-}\beta)\, rT\big), \quad k_{\text{bg}} = \text{round}\big(\beta\, rT\big). \tag{6}$$

Unlabeled fallback uses spectral flux binarized at $\rho{=}0.30$. Budget underflow clamps to available frames per region. Contiguous blocks of width $w \in \{4, 16\}$ are sampled independently inside each region,

$$\mathcal{M}_{\text{EA}} = \mathcal{M}_{\text{evt}} \cup \mathcal{M}_{\text{bg}},$$
$$\mathcal{M}_{\text{evt}} \subseteq \{t : e_t{=}1\}, \quad \mathcal{M}_{\text{bg}} \subseteq \{t : e_t{=}0\}. \tag{7}$$

ERB (energy-reweighted block). Plain block masking spends a large share of its budget on silent frames that are easy to reconstruct Seth et al. (2024); Niizumi et al. (2026). ERB addresses this by keeping block geometry $w \in \{4, 16\}$ but excluding silence from sampling. Mel-energy partitions frames into a silence set S and its complement with a 40 dB VAD threshold pattern. We sample blocks among non-silent frames until $|\mathcal{M}|/T \approx r$. Underflow clamps to available frames; all-silence clips fall back to uniform weighting.

4.3 Mask Curricula

We also study curricula that change mask difficulty or structure across PT1 epochs. All three strategies keep a block geometry and differ in how the mask index set evolves.

EH-MAM. We adapt EH-MAM Seth et al. (2024) as an easy-to-hard *placement* curriculum under fixed block width. Contrary to standard block masking, where blocks are placed randomly, EH-MAM places blocks at frames with higher loss values. In addition, EH-MAM can be partially overlapping. Early epochs sample mostly random blocks. Later epochs increasingly prefer currently hard (high-loss) blocks. The hard share follows $\kappa(e) = \min\big((e+1)/E,\, 0.8\big)$ at epoch e of E PT1 epochs.

HPM. Our HPM-inspired hard-mining curriculum Wang et al. (2023) keeps the same easy-to-hard placement idea, but scores difficulty with a frame-level EMA of reconstruction MSE, rather than a relative loss predictor,

$$d_t \leftarrow \gamma\, d_t + (1-\gamma)\, \ell_t, \qquad \gamma{=}0.9, \tag{8}$$

and places w-wide spans at high-d_t positions within an inverse-block host, so the short-gap structure of InvBlock is preserved but gaps are biased toward hard frames rather than drawn uniformly. The realized mask budget is lower than the nominal $r{=}0.75$ (reported setting, $r_{\text{eff}} \approx 0.35$).

HierBlock. HierBlock schedules block *width* rather than placement. It uses $w{=}4$ for the first half of PT1 and switches to $w{=}32$ for the second half. This is a simplified geometry proxy inspired by SemMAE's curriculum Li et al. (2022).

5 Results

In this section, we discuss our experimental results. Our main results are presented in Table 2. Our Transformer-XL baseline is an adaptation of MAT-SED for the present ablation study. It obtains a PSDS1 of 0.531, compared with the published MAT-SED result of 0.587 Cai et al. (2024b). The protocols differ: our system includes a parallel CNN branch and uses block width $w{=}4$, whereas the original uses a pure Transformer architecture and block width $w{=}10$. The training schedules also differ. We therefore interpret the masking and decoder results as comparisons within our experimental pipeline, rather than as a reproduction of the published MAT-SED score. The best PSDS_1 of $.540$ is achieved by Block ($w{=}4$) with a BiGRU or BiMamba decoder. The board can be lifted to .550-.570 and higher with little change to our environment, using more advanced post-processing methods (i.e., cSEBBs Ebbers et al. (2024)), and extended FT2 training schedules.

5.1 Geometry-based masks

Geometry-based masks are standard in previous literature. We compared them using a fixed TransformerXL decoder. In our case, they consistently improved PSDS_1 but decreased eF1 compared to the baseline without SSL. This shows that geometry-based masks can improve the temporal localization of events, as measured by PSDS_1, to the detriment of the stricter detection collars applied to eF1. Figure 1 plots class-specific PSDS_1 scores for different block widths using block masking. Each row's best PSDS_1 width is highlighted in white. *Dishes* proved to be the most difficult class, having the lowest PSDS_1 value. As shown in Table 1, this class has the lowest average duration ($\bar{d} = 0.54\,\text{s}$) and simultaneously the highest amount of overlap with one (51.4%) or multiple events (13.3%). Overall, most other classes favor wider blocks ($w{\geq}32$), while short transient classes (*Dog*, *Cat*) peak at smaller widths ($w{\leq}16$), consistent with block width acting as a proxy for target event duration.

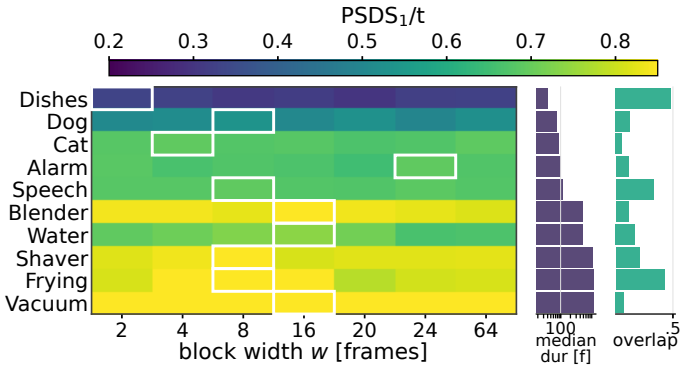


Figure 1: Class-wise PSDS_1 heatmap for different block widths under standard block masking. The data used is the DESED real validation set. White outlines mark the best width per class.

	Real train (3373)			Real val (1168)		
Class	$\bar{d}$	ov_1	$\text{ov}_{>1}$	$\bar{d}$	ov_1	$\text{ov}_{>1}$
Alarm bell	2.17	26.8%	0.3%	1.96	10.9%	0.4%
Blender	5.22	22.2%	0.0%	5.23	11.0%	0.0%
Cat	0.89	22.8%	0.8%	1.38	5.4%	0.0%
Dishes	0.54	51.4%	13.3%	0.63	41.8%	6.7%
Dog	0.84	26.4%	0.5%	1.41	12.3%	0.1%
Elec. shaver	6.85	50.1%	0.5%	7.73	20.7%	0.0%
Frying	8.26	46.0%	3.5%	8.26	39.9%	2.7%
Running water	6.45	35.9%	0.5%	5.25	16.5%	0.5%
Speech	1.62	43.9%	1.3%	1.50	32.1%	1.0%
Vacuum	7.19	27.8%	0.0%	8.48	7.4%	0.0%
All	2.01	39.6%	1.6%	2.10	21.8%	0.9%

Table 1: Event duration and overlap on DESED real data. $\bar{d}$ is the mean annotated event duration in seconds. ov_1 is the percentage of each event's duration that overlaps with exactly one other event; $\text{ov}_{>1}$ is the percentage that overlaps with more than one other event.

5.2 Saliency-based masks

We evaluate three proposed saliency-based masks (TEM, EA, ERB) and AttMask as a vision-derived baseline. TEM and EA lead eF1 among masking strategies (.617 / .609), but trail the best PSDS_1 entries. AttMask and ERB are stronger on PSDS_1 (.535 / .533), marginally outperforming geometry-based masks. Guided placement therefore moves the operating point along the PSDS_1 / eF1 axis rather than dominating both metrics. Focusing the sampling on regions where the input tokens show larger activations, as these masking strategies do, improves overall performance. We hypothesize that their focus on highly-active frames forces the decoder to reconstruct regions with higher event overlap, or higher event activity, thus learning representations that help distinguish different events. Nevertheless, future work should try to isolate what these activations encode, especially in the case of TEM which relies on PaSST activations.

5.3 Mask Curricula

Changing the masking difficulty or structure across epochs across PT1 epochs improves downstream SED. Effects were less consistent than for fixed geometry. EH-MAM sorts frames by reconstruction loss and uses an easy-to-hard placement curriculum $\kappa(e) = \min((e+1)/E,\, 0.8)$ to transition from random masks (early epochs) to hard (high-loss) placement. This schedule alone reduces both PSDS_1 and eF1 relative to no-SSL (.516 vs. .523). HPM applies the same loss-based curriculum as EH-MAM but uses InvBlock geometry, placing blocks at high-loss frame positions. Realized mask budget drops to $r_{\text{eff}} \approx 0.35$ due to hard-region clustering. This focuses pretraining on hard regions, raising PSDS_1

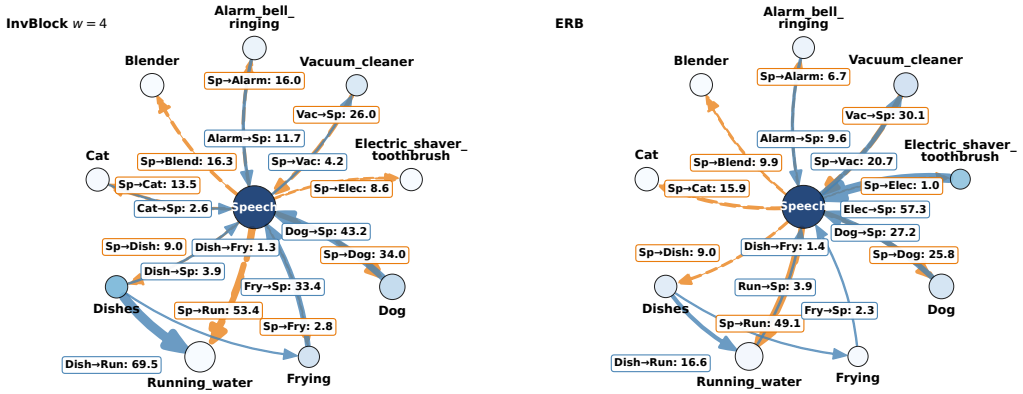
Variant	Params	$PSDS_1$ [CI_{95}]	$PSDS_2$	eF1	MSE
Masking strategies (w. TransformerXL decoder)					
No SSL	12.4M	.523 [.497, .552]	.751	.595	–
Geometry-based					
Random	12.4M	.525 [.503, .551]	.766	.576	.009
Block (w=4)	12.4M	.531 [.510, .559]	.764	.587	.009
InvBlock (w=4)	12.4M	.532 [.505, .561]	.756	.577	.007
Saliency-based					
TEM	12.4M	.521 [.492, .552]	.755	.617	.014
EA	12.4M	.531 [.505, .562]	.766	.609	.022
ERB	12.4M	.533 [.507, .563]	.752	.573	.006
AttMask	12.4M	.535 [.511, .562]	.756	.582	.006
Curriculum-based					
EH-MAM	12.4M	.516 [.488, .548]	.762	.591	.006
HierBlock	12.4M	.522 [.494, .552]	.742	.603	.007
HPM	12.4M	.533 [.509, .559]	.757	.585	.005
Decoders (w. block masking)					
Transformer					
Conformer	21.4M	.495 [.464, .517]	.713	.586	.007
Transformer	10.6M	.524 [.496, .553]	.773	.582	.014
TransformerXL	12.4M	.531 [.510, .559]	.764	.587	.009
RNN					
BiLSTM	37.8M	.529 [.503, .559]	.746	.559	.013
BiGRU	28.3M	**.540** [.511, .568]	.772	**.633**	.013
SSM					
Hydra	11.5M	.529 [.504, .560]	.752	.571	.012
Mamba	11.3M	.534 [.507, .563]	.765	.570	.012
BiMamba	22.6M	**.540** [.512, .569]	.765	.569	.009
xLSTM					
xLSTM	10.8M	.536 [.504, .564]	.765	.560	.009

Table 2: Masking strategies on a fixed decoder and decoder architectures on fixed block mask (w=4, r=0.75). Dataset: DESED real validation (seed 21). CI_{95} is a clip bootstrap (n=1000). $PSDS_2$ is a point estimate at the best-$PSDS_1$ checkpoint.

(and *Dishes*) but not collar-based eF1. HierBlock switches from w=4 to w=32 halfway through PT1, yet underperforms fixed Block w=4 (.531) and w=32 (.533, cf. 1) on validation $PSDS_1$.

5.4 Decoder Architectures

Previous masking results used TransformerXL. Historically, SED pipelines used BiRNNs as their temporal decoder Shao et al. (2023). However, different architectures carry different inductive biases which can influence downstream SED performance. In Table 2, we compare decoder architectures under standard block masking with a default width of w=4, however parameter counts are not matched across families. Transformer blocks for downstream SED were introduced in MAT-SED to succeed the RNN-based DCASE network baseline. In that setting, TransformerXL was preferred through shift-invariant relative position attention (RPE). In our sweep, the same pattern holds within the attention family. TransformerXL reaches $PSDS_1$=.531, ahead of a plain Transformer (.524) and Conformer (.495). Conformer, which couples self-attention with local depthwise convolutions, is weakest on $PSDS_1$ despite competitive collar-based event F1. BiMamba (.540) and Mamba (.534) exceed Transformer-XL (.531), while Hydra (.529) remains competitive at a lower parameter count (11.5M). This illustrates how the ability to handle longer context without the quadratic cost of transformers can be highly beneficial for SED. A BiGRU decoder matches BiMamba on $PSDS_1$ (.540) but achieves substantially higher collar-based eF1, albeit for a higher parameter count. This



(a) InvBlock (w=4). (b) ERB.

Figure 2: Mass transfer graphs for InvBlock (w=4) and ERB. Node color encodes dominance (total stolen mass). Arrows point from interferer to victim; color distinguishes the two directions.

reveals that SSMs efficiently model temporal dependencies but sacrifice on temporal precision that collar-based F1 enforces. An xLSTM decoder Beck et al. (2024) built from bidirectional mLSTM layers (alternating forward/backward scan, expand factor 2, L=3) reaches PSDS_1=.536 at 10.8M parameters. Future work could study decoder-mask interactions and the decoupling between reconstruction error and frame-wise detection metrics more directly.

5.5 Domain bias and Polyphonicity

Detection quality depends strongly on separating overlapping events, especially short ones (Figure 1). We analyze this with a *mass transfer graph* over clip-level ownership (Appendix F): each clip has an annotated and a predicted owner class, and edges quantify how co-occurring classes interfere with ownership. Comparing InvBlock (w=4) and ERB (Figure 2), three patterns stand out. ***Speech* is the major interaction hub** under both samplers, matching its duration and overlap dominance (Figure 5, Table 1). ***Speech* vulnerability is sampler-dependent**: ERB elevates *Electric shaver*→*Speech* and suppresses *Frying*→*Speech* relative to InvBlock. **Kitchen events form a separate cluster**, sharper under InvBlock than ERB. Appendix E further shows that short, overlapped events remain hardest across mask families, while long events converge (Figures 5 and 6).

6 Conclusion

Our work introduces one of the largest recent ablations for different SED recipes. Using a rigorous evaluation framework, we were able to show that using temporal decoders with a standard masking strategy can surpass more complicated masking strategies. Finally, our in-depth evaluation shows that shorter events that occur in more polyphonic setups still remain challenging for SED pipelines, and that training set bias propagates to the behavior of the detector.

Limitations. Cross-dataset alignment on long-form strong labels (e. g., AudioSet Strong) are left for future work. Decoder capacity is only partially matched (Appendix Table 4). InvBlock and HPM also realize lower mask budgets than nominal Block at r=0.75, so geometry and corruption rate remain partially confounded without a budget-matched Block control. The mass transfer graph analysis (Section 5.5) assigns clip ownership to the longest-duration class, which introduces a duration bias.

Our work stands to inform future SED development: In a domain where scaling remains hard due to the inherent difficulty of the task, it is important to develop new architectures and new training strategies that improve performance. Our new evaluation metrics can additionally help to characterize these different recipes and guide future development.

AI USE STATEMENT

In this work, we have not used generative AI tools for any tasks with required disclosure under the ICLR 2027 AI Policy for Authors including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating or proving mathematical claims, proposing or refining hypotheses, designing research methodology or experiments, implementing methods, translation, dataset cleaning, qualitative analysis, or interpreting results. Those required-disclosure tasks are not applicable beyond that abstention. However, we used generative AI tools for recommended-disclosure tasks: accelerating iteration on scientific figures and images, and peer-reviewing research code for correctness and clarity. All AI-assisted figure edits and code-review suggestions were checked by the authors before inclusion. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our experiments relied on publicly available datasets. We are aware of the risk of publishing methods that can be used for surveillance using this public data. We ensure that the distribution of our code and models is handled with caution and ethical consideration.

REPRODUCIBILITY STATEMENT

We document the implementation details for reproducing our experiments in Section 3 and in Appendix A as well as the supplementary code files that are included with this submission.

REFERENCES

Alan Baade, Puyuan Peng, and David Harwath. MAE-AST: Masked autoencoding audio spectrogram transformer, 2022. URL https://arxiv.org/abs/2203.16691.

Alexei Baevski, Arun Babu, Wei-Ning Hsu, and Michael Auli. Efficient self-supervised learning with contextualized target representations for vision, speech and language, 2023. URL https://arxiv.org/abs/2212.07525.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. BEiT: BERT pre-training of image transformers, 2022. URL https://arxiv.org/abs/2106.08254.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xLSTM: Extended long short-term memory. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2405.04517.

Çağdaş Bilen, Giacomo Ferroni, Francesco Tuveri, Juan Azcarreta, and Sacha Krstulović. A framework for the robust evaluation of sound event detection. In *Proc. ICASSP*, pp. 61–65. IEEE, 2020.

Pengfei Cai, Yan Song, Nan Jiang, Qing Gu, and Ian McLoughlin. Prototype based masked audio model for self-supervised learning of sound event detection, 2024a. URL https://arxiv.org/abs/2409.17656.

Pengfei Cai, Yan Song, Kang Li, Haoyu Song, and Ian McLoughlin. MAT-SED: A masked audio transformer with masked-reconstruction based pre-training for sound event detection, 2024b. URL https://arxiv.org/abs/2408.08673. version: 2.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2978–2988, 2019. doi: 10.18653/v1/P19-1285. URL https://arxiv.org/abs/1901.02860.

Virginia De Sa. Learning classification with unlabeled data. *Advances in neural information processing systems*, 6, 1993.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

Janek Ebbers, François G. Germain, Gordon Wichern, and Jonathan Le Roux. Sound event bounding boxes. In *Interspeech 2024*, pp. 562–566, 2024. doi: 10.21437/Interspeech.2024-2075. URL https://www.isca-archive.org/interspeech_2024/ebbers24_interspeech.html.

Wenmiao Gao and Han Yin. Enhancing stereo sound event detection with bimamba and pretrained PSELDNet. In *2025 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 2020–2025, 2025. doi: 10.1109/APSIPAASC65261.2025.11249073. URL https://ieeexplore.ieee.org/document/11249073.

Yuan Gong, Cheng-I. Jeff Lai, Yu-An Chung, and James Glass. SSAST: Self-supervised audio spectrogram transformer, 2022. URL https://arxiv.org/abs/2110.09784.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces, 2024. URL https://arxiv.org/abs/2312.00752.

Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang. Conformer: Convolution-augmented transformer for speech recognition, 2020. URL https://arxiv.org/abs/2005.08100.

Manu Harju, Irene Martin Morato, and Annamaria Mesaros. Does paid crowdsourcing still pay off? sifting through annotator noise in crowdsourced audio labels. In *Workshop on Detection and Classification of Acoustic Scenes and Events*, pp. 56–60. DCASE, 2024.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, 2022. doi: 10.1109/CVPR52688.2022.01553. URL https://arxiv.org/abs/2111.06377.

Sukjun Hwang, Ali Behrouz, and Albert Gu. Hydra: Bidirectional state space models through generalized matrix mixers, 2024. URL https://arxiv.org/abs/2407.09941.

Ioannis Kakogeorgiou, Spyros Gidaris, Bill Psomas, Yannis Avrithis, Andrei Bursuc, Konstantinos Karantzalos, and Nikos Komodakis. What to hide from your students: Attention-guided masked image modeling. In *European Conference on Computer Vision (ECCV)*, volume 13690 of *Lecture Notes in Computer Science*, pp. 300–318, 2022. doi: 10.1007/978-3-031-20056-4_18. URL https://arxiv.org/abs/2203.12719.

Gang Li, Heliang Zheng, Daqing Liu, Chaoyue Wang, Bing Su, and Changwen Zheng. SemMAE: Semantic-guided masking for learning masked autoencoders, 2022. URL https://arxiv.org/abs/2206.10207.

Shuo Liu, Adria Mallol-Ragolta, Emilia Parada-Cabeleiro, Kun Qian, Xin Jing, Alexander Kathan, Bin Hu, and Bjoern W. Schuller. Audio self-supervised learning: A survey, 2022. URL https://arxiv.org/abs/2203.01205.

David Marr. Vision: A computational investigation into the human representation and processing of visual information, 1982.

Irene Martín-Morató and Annamaria Mesaros. Strong labeling of sound events using crowdsourced weak labels and annotator competence estimation. *IEEE/ACM transactions on audio, speech, and language processing*, 31:902–914, 2023.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. Metrics for polyphonic sound event detection. *Applied Sciences*, 6(6):162, 2016. doi: 10.3390/app6060162. URL https://www.mdpi.com/2076-3417/6/6/162.

Annamaria Mesaros, Toni Heittola, Tuomas Virtanen, and Mark D Plumbley. Sound event detection: A tutorial. *IEEE Signal Processing Magazine*, 38(5):67–83, 2021.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

Hyeonuk Nam and Yong-Hwa Park. JiTTER: Jigsaw temporal transformer for event reconstruction for self-supervised sound event detection, 2025. URL https://arxiv.org/abs/2502.20857. version: 1.

Daisuke Niizumi, Daiki Takeuchi, Masahiro Yasuda, Binh Thien Nguyen, Noboru Harada, and Nobutaka Ono. Rethinking Masking Strategies for Masked Prediction-based Audio Self-supervised Learning, mar 2026. URL http://arxiv.org/abs/2603.23810. arXiv:2603.23810 [eess] version: 1.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, 2024.

Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le. SpecAugment: A simple data augmentation method for automatic speech recognition. In *Interspeech 2019*, pp. 2613–2617, 2019. doi: 10.21437/Interspeech.2019-2680. URL https://arxiv.org/abs/1904.08779.

Ashish Seth, Ramaneswaran Selvakumar, S Sakshi, Sonal Kumar, Sreyan Ghosh, and Dinesh Manocha. EH-MAM: Easy-to-hard masked acoustic modeling for self-supervised speech representation learning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 6386–6400, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.366. URL https://aclanthology.org/2024.emnlp-main.366/.

Nian Shao, Xian Li, and Xiaofei Li. Fine-tune the pretrained ATST model for sound event detection, 2023. URL https://arxiv.org/abs/2309.08153. version: 2.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, 2018. URL https://arxiv.org/abs/1703.01780.

Andreas Triantafyllopoulos, Iosif Tsangko, Alexander Gebhard, Annamaria Mesaros, Tuomas Virtanen, and Björn W. Schuller. Computer audition: From task-specific machine learning to foundation models. *Proceedings of the IEEE*, 113(4):317–343, 2025. doi: 10.1109/JPROC.2025.3593952.

Nicolas Turpault, Romain Serizel, Justin Salamon, and Ankit Parag Shah. Sound event detection in domestic environments with weakly labeled data and soundscape synthesis. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019)*, pp. 253–257. New York University, 2019. ISBN 978-0-578-59596-2. doi: 10.33682/006b-jx26. URL http://hdl.handle.net/2451/60771.

Haochen Wang, Kaiyou Song, Junsong Fan, Yuxi Wang, Jin Xie, and Zhaoxiang Zhang. Hard Patches Mining for Masked Image Modeling, apr 2023. URL http://arxiv.org/abs/2304.05919. arXiv:2304.05919 [cs].

A Environment

All experiments run on a single NVIDIA RTX 3090 GPU with 24 GB of VRAM, which allows FT2 batches of the form $(2, 2, 2, 2)$ for the full-parameter model, whereas pretraining uses batch shape of the form $(4, 4, 4, 4)$. The implementation uses PyTorch 2.8 with CUDA 12.8 under Python 3.10. Unless noted, the training seed is 21. A full three-stage training takes about 14 hours of wall-clock time on this hardware. We provide documentation of training configuration and code in the supplementary material to allow researchers to build on this set of ablations to elevate future research in building a robust and self-supervised sound event detection pipeline.

Table 3: AttMask hyperparameter sweep on DESED real validation (teacher). MSE is the PT1 reconstruction loss. ρ controls the high-attention pool size via $\max(rT, \lfloor \rho T \rfloor)$, and L is the number of early PaSST attention layers averaged into a_t. Placement is unstructured over frames. No block width is used.

Sampler	$PSDS_1$	$PSDS_2$	MSE
AttMask (ρ=0.30, L=6)	.534	.758	.006
AttMask (ρ=0.50, L=6)	**.535**	.759	.006
AttMask (ρ=0.75, L=6)	.520	.759	.006
AttMask (ρ=0.50, L=2)	.520	.765	.006
AttMask (ρ=0.50, L=4)	.521	.766	.006
AttMask (ρ=0.50, L=10)	.518	.764	.027

B Decoder hyperparameter ablation

B.1 AttMask hyperparameter ablation

Table 3 reports a small AttMask sweep on DESED real validation (teacher), complementary to the saliency row of Table 2. As in Section 4, PaSST self-attention maps from the first L layers are averaged to a per-frame score a_t, and a high-attention pool of size $\max(rT, \lfloor \rho T \rfloor)$ is formed from the largest a_t values. Masks are then sampled as independent frames inside that pool. No contiguous block width is used. The table also includes ρ=0.75 and additional L settings beyond the reported operating points $\rho \in \{0.30, 0.50\}$ and L=6. MSE is the PT1 reconstruction loss on masked tokens. The best $PSDS_1$ of .535 uses ρ=0.50 and L=6, matching the AttMask $PSDS_1$ in Table 2. Raising ρ to 0.75 drops $PSDS_1$ to .520, so over-expanding the attention pool hurts more than it helps. With ρ=0.50, increasing L from 2 to 6 raises $PSDS_1$ from .520 to .535, while L=10 raises MSE sharply (.027) and does not improve $PSDS_1$.

Overall, AttMask is moderately sensitive to the attention-pool size and the depth of layers used for placement. The reported operating point (ρ=0.50, L=6) achieves $PSDS_1$=.535, but most nearby hyperparameter choices fall below or near the No-SSL baseline, indicating this method requires careful tuning to exceed the baseline.

B.2 BiMamba hyperparameter ablation

Table 4 ablates the BiMamba decoder under fixed block masking on DESED real validation (teacher), complementary to the SSM row of Table 2. Unless noted, runs use w=4, r=0.75, L=3, d_{state}=16, d_{conv}=4, and sinusoidal positional encoding. MSE is the PT1 reconstruction loss. The default setting matches the BiMamba entry in Table 2 ($PSDS_1 = 0.540$). Reducing depth to L=2 keeps $PSDS_1$ close (.535) at lower capacity (15.1M), while L=4 raises eF1 (.585) but not $PSDS_1$. Larger SSM state ($d_{\text{state}} \in \{64, 128\}$) and wider depthwise convolutions ($d_{\text{conv}} \in \{8, 16\}$) do not improve $PSDS_1$ over the default. Learnable positional encodings underperform sincos at both w=4 and w=16. Widening the PT1 block to w=16 matches default $PSDS_1$ (.540) with slightly higher eF1 and higher MSE. Overall, the reported BiMamba operating point is stable: nearby capacity and geometry changes move eF1 and MSE more than $PSDS_1$.

B.3 xLSTM hyperparameter ablation

Table 5 ablates the xLSTM decoder under fixed block masking on DESED real validation (teacher), complementary to the xLSTM row of Table 2. Unless noted, runs use w=4, r=0.75, L=3, expand factor 2, alternating mLSTM/sLSTM blocks, and sinusoidal positional encoding. MSE is the PT1 reconstruction loss. The default setting matches the xLSTM entry in Table 2 ($PSDS_1$ .536). Reducing depth to L=2 cuts capacity (7.2M) and drops $PSDS_1$ to .518. L=4 raises eF1 (.576) but not $PSDS_1$. Learnable positional encodings underperform sincos on $PSDS_1$ (.519 vs. .536) while raising eF1 slightly. Overall, the reported xLSTM operating point is the strongest among the finished depth and position ablations, and nearby changes move eF1 and MSE more than they improve $PSDS_1$.

Table 4: BiMamba decoder ablations on DESED real validation (teacher). Block masking with w=4, r=0.75 unless noted. MSE is the PT1 reconstruction loss. Default: L=3, d_{state}=16, d_{conv}=4, sincos.

Variant	Params	PSDS_1	eF1	MSE
BiMamba (default)	22.6M	**.540**	.569	.009
BiMamba (L=2)	15.1M	.535	.552	.007
BiMamba (L=4)	30.2M	.531	.585	.007
BiMamba (d_{state}=64)	24.0M	.530	.573	.008
BiMamba (d_{state}=128)	25.7M	.530	.565	.009
BiMamba (d_{conv}=8)	22.7M	.535	.578	.008
BiMamba (d_{conv}=16)	22.7M	.527	.569	.008
BiMamba (learnable, w=4)	23.4M	.530	.576	.009
BiMamba (learnable, w=16)	23.4M	.530	.577	.011
BiMamba (w=16, sincos)	22.6M	**.540**	.577	.012

Table 5: xLSTM decoder ablations on DESED real validation (teacher). Block masking with w=4, r=0.75 unless noted. MSE is the PT1 reconstruction loss. Default: L=3, expand 2, alternating, sincos.

Variant	Params	PSDS_1	eF1	MSE
xLSTM (default)	10.8M	**.536**	.560	.009
xLSTM (L=2)	7.2M	.518	.548	.008
xLSTM (L=4)	14.4M	.528	.576	.009
xLSTM (learnable)	11.6M	.519	.565	.006

C CLASS-WISE DECODER COMPARISON

Figure 3 expands the decoder rows of Table 2 to per-class PSDS_1 on DESED real validation (teacher), under fixed block masking (w=4, r=0.75). Bars show Transformer (TF), Transformer-XL (TXL), Conformer (Conf), Mamba, and BiMamba. Class order follows the plot abbreviations (*Alarm*, *Blender*, *Cat*, *Dishes*, *Dog*, *Electric toothbrush*, *Frying*, *Running water*, *Speech*, *Vacuum cleaner*). Aggregate PSDS_1 alone hides where the family ranking is decided. Conformer is weakest on almost every class and drops sharply on *Dishes*, *Dog*, and *Speech*, which matches its low clip-level PSDS_1 (.495) despite competitive collar eF1 in Table 2. Within the attention family, Transformer-XL sits above plain Transformer on most classes, consistent with the relative-position bias discussed in Section 5. Mamba and BiMamba lead or match the attention baselines on the hard short classes (*Dishes*, *Dog*) and stay close on long, high-scoring classes (*Electric toothbrush*, *Vacuum cleaner*), where all heads except Conformer sit near .8–.95. *Dishes* remains the lowest absolute band for every decoder (roughly .15–.35), so architecture choice does not remove the short, overlapped kitchen regime highlighted in Appendix E. Overall, the SSM advantage in Table 2 is not a single-class artifact. It appears as a broad lift on difficult classes, while long stationary events already score high under most decoders.

D MASK GEOMETRY ILLUSTRATION

Figure 4 shows how geometry- and saliency-based samplers place corruption on one 10 s DESED clip. The top panel is the mel spectrogram. Each lower row is the per-frame mask probability $P(\text{masked})$ under a fixed sampler (darker = more likely masked). Random draws independent frames and yields a fine salt-and-pepper pattern. Block (w=4) replaces that pattern with contiguous runs of length w. InvBlock (w=4) inverts the block geometry. Most of the clip stays visible, and only short gaps are corrupted, matching the lower realized mask rate discussed in Section 4. TEM concentrates mass on high-activation / high-change frames and tracks transient structure in the spectrogram. ERB keeps block width but excludes silence, so long active regions receive near-certain masks while quiet spans stay unmasked. EA (GT) uses strong-label event unions. EA (unlab.) uses the flux/energy fallback on unlabeled material, producing thinner, more fragmented event support than the GT row. The figure is qualitative. It is meant to make the geometric and saliency differences in Section 4 concrete before the duration-binned comparisons in Appendix E.

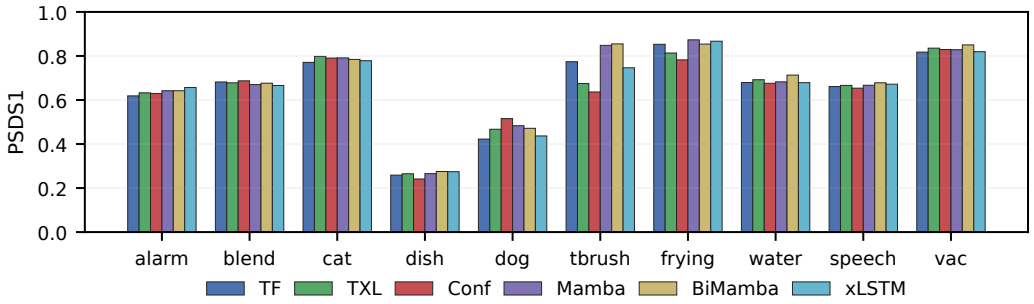


Figure 3: Class-wise PSDS_1 on DESED real validation (teacher) for decoder architectures under block masking (w=4, r=0.75). Complementary to the decoder block of Table 2. Abbreviations: TF = Transformer, TXL = Transformer-XL, Conf = Conformer.

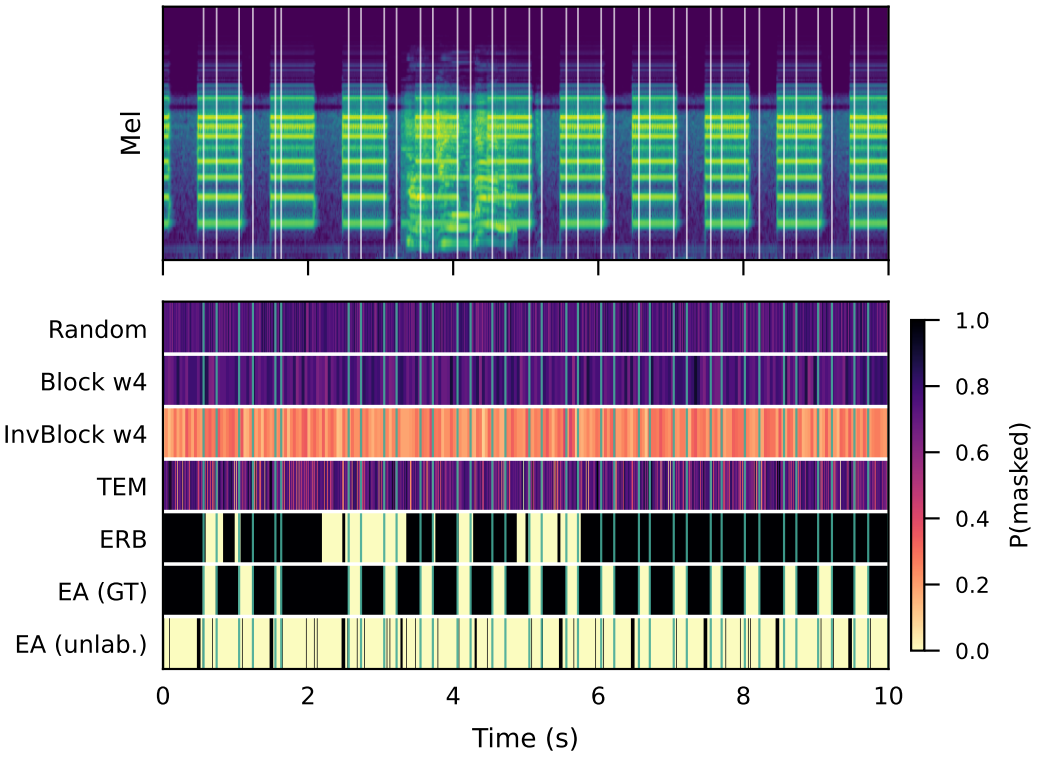


Figure 4: Mask placement on a single DESED clip. Top: mel spectrogram. Bottom rows: per-frame $P(\text{masked})$ for Random, Block (w=4), InvBlock (w=4), TEM, ERB, and EA under ground-truth vs. unlabeled event proxies. Darker frames are more likely to be corrupted.

E Extended mask and duration analysis

Figure 5 summarizes class duration and pairwise overlap on real training data. *Speech* dominates total mass (623 min) and co-occurs with every class, while *Dishes* is short (65 min) and heavily overlapping (41% with *Frying*, 37% with *Speech*). This kitchen pair, plus ubiquitous *Speech*, helps explain why short events remain hard even when aggregate PSDS_1 looks competitive. Figure 6 disaggregates validation PSDS_1 by event-duration bin. Across mask families, PSDS_1 rises with duration: worst on $[0, 1)$ s and best on $[9, 10)$ s (cf. Figure 1, Table 1). Sampler differences matter most in the short- and mid-duration regime. On $[0, 1)$ s, AttMask / EA / ERB sit near .57-.58, while

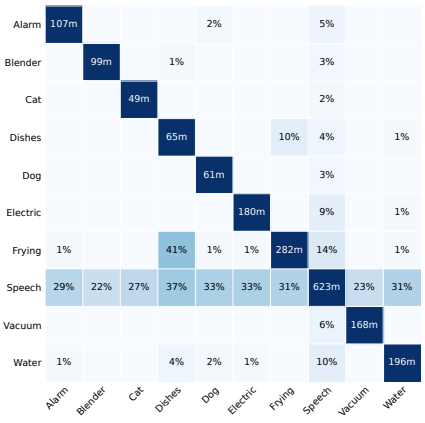


Figure 5: Event duration and overlap on DESED strongly-labeled training data (real and synthetic). Diagonal shows total duration of each event in minutes. Off-diagonal entries show the percentage of the column class's total duration that overlaps with the row class.

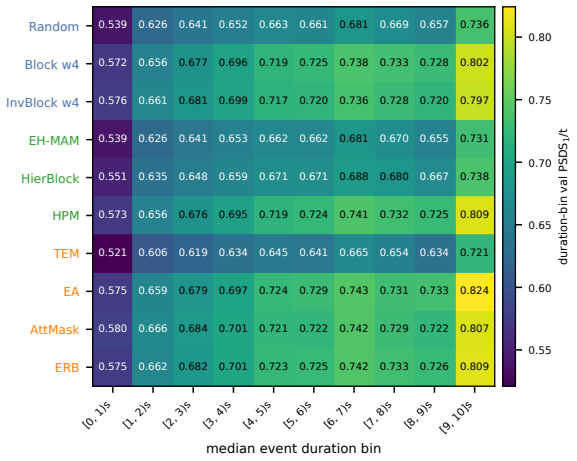


Figure 6: Bin-wise, PSDS_1 for geometry, curriculum, and saliency-based masks. Geometry, curriculum, and saliency-based groups are colored in blue, green, and orange respectively.

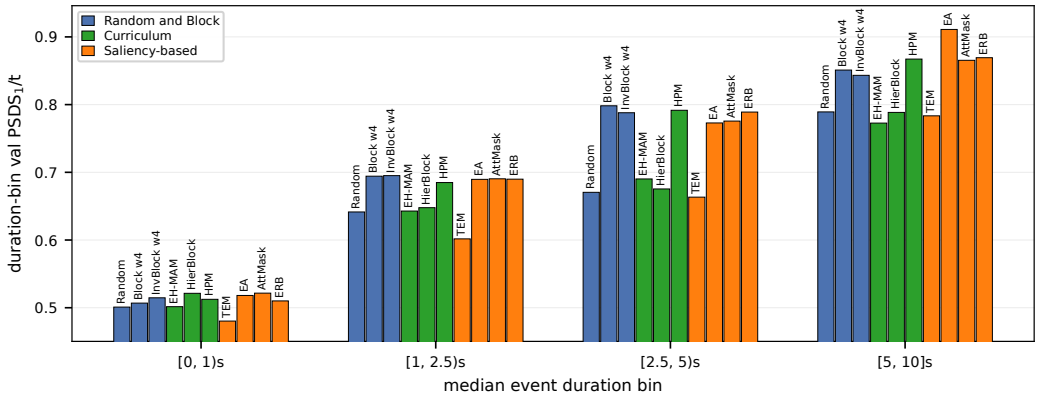


Figure 7: Duration-binned validation PSDS_1 on the real validation set. Each class is assigned to the bin of its median event duration. Bars are absolute per-bin scores for the mask families in Table 2. Color denotes the mask group (geometry, curriculum, saliency).

TEM falls to about .52. Toward $[9, 10)$ s, the stronger rows sit above .78, while TEM, EH-MAM, and HierBlock remain lower.

Section 5 shows that short, overlapped events such as *Dishes* remain hard under all mask families (Table 1, Figures 1 and 6). The figures below expand that analysis on DESED real validation (teacher). Median-duration panels assign each class to the bin of its median event duration and report duration-bin PSDS_1 for the mask families in Table 2 (geometry, curriculum, and saliency). True-duration panels instead bin individual ground-truth events by their length on the same DESED real validation split. Colors follow the main paper: geometry (blue), curriculum (green), and saliency (orange). Figure legends label the orange group as Salience-based.

Figure 7 aggregates the same mask families as Figure 6 into four median-duration bins. Absolute scores rise monotonically from $[0, 1)$ s to $[5, 10]$ s for every sampler. Differences among masks are largest in the mid-duration bins, where Block (w=4), InvBlock (w=4), HPM, and several saliency methods pull ahead of Random and EH-MAM. On $[5, 10]$ s, EA is strongest, while TEM remains the weakest saliency entry across bins. This matches the main-text claim that mask choice matters most

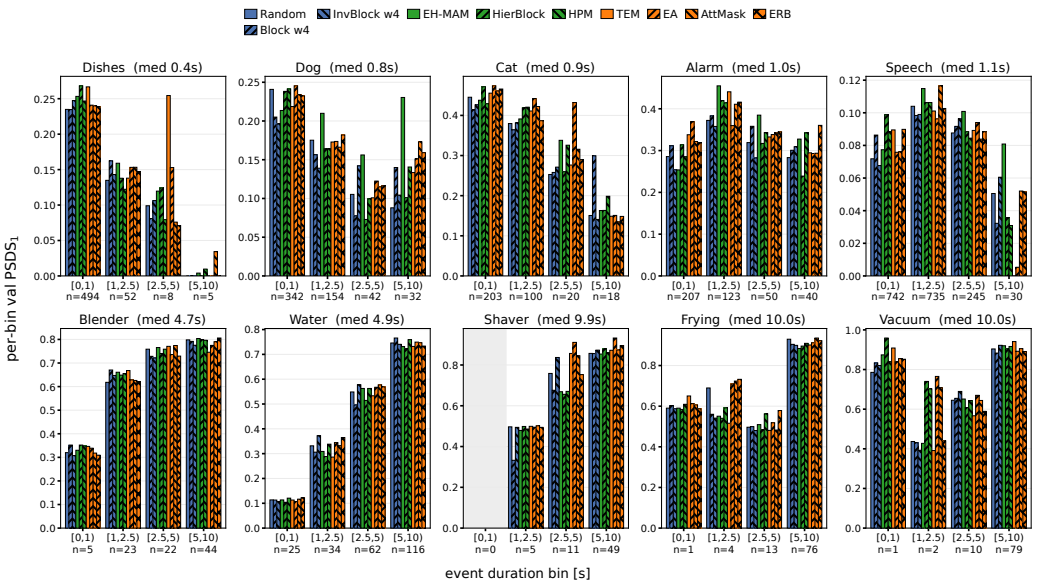


Figure 8: True duration-binned validation PSDS_1 on DESED real validation (teacher). One panel per class, ordered by median event duration. Bar height is the absolute per-bin score. n under each bin is the number of ground-truth events.

where short and mid-length events live, and that long events converge to high scores regardless of sampler.

Figure 8 conditions on true event length rather than class median, still on real validation. Long-dominated classes (*Running water*, *Frying*, *Vacuum*) rise strongly toward $[5, 10)$ s. Short classes behave differently: *Dishes* peaks on $[0, 1)$ s (n=494) but only at a low absolute level, then falls in longer bins, consistent with its short mean duration and high overlap in Table 1. *Speech* stays low across bins despite large n. Sparse short bins for appliance classes (small n, including n=0 for *Electric shaver* on $[0, 1)$ s) should be read cautiously. Sampler gaps shrink once events occupy most of the clip, again matching Figure 6.

Figure 9 subtracts the bin mean so that positive bars mark masks that beat the local average. No single sampler dominates every class and bin. Saliency methods are competitive on short *Dishes* ($[0, 1)$ s), while TEM spikes on longer *Cat* bins. Geometry and curriculum entries can lead in selected cells, including some low-n long bins of short classes, where swings are large. This validation view supports the paper claim that mask choice is most consequential for short and mid-duration regimes, and that aggregate PSDS_1 alone hides class- and duration-specific winners.

F Mass transfer graph

Section 5.5 uses the mass transfer graph to compare ownership interactions under InvBlock (w=4) and ERB (Figure 2). We give the formal construction here. *Speech* provides the main hub context: it has the largest annotated duration on real training data (Figure 5) and high pairwise overlap on both real train and real validation (Table 1).

Definition (Mass Transfer Graph). Let $\mathcal{S}$ be the evaluation clips (DESED real validation) and $\mathcal{Y}$ the set of event classes. For a clip $s \in \mathcal{S}$ with active labels $Y(s) \subseteq \mathcal{Y}$, let $d_c(s)$ be the total annotated duration of class c in s (consistent with mean durations $\bar{d}$ in Table 1). The annotated owner is

$$o(s) = \arg \max_{c \in Y(s)} d_c(s). \tag{9}$$

The predicted owner $\hat{o}(s)$ is defined analogously from predicted event durations after frame-score thresholding at the teacher operating point used for validation scoring. For each owner class $i \in \mathcal{Y}$,

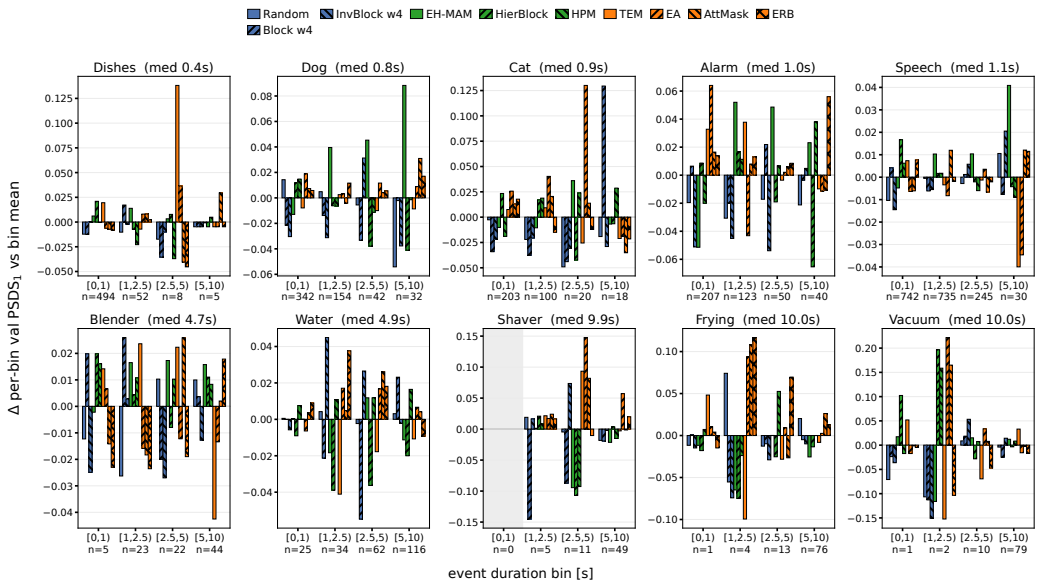


Figure 9: True duration-binned validation PSDS_1 on DESED real validation (teacher), shown as Δ relative to the per-bin mean across masks. One panel per class, ordered by median event duration. n under each bin is the number of ground-truth events.

let $N_i = |\{s \in \mathcal{S} : o(s) = i\}|$. For each co-occurring class $k \neq i$, define the conditioned subsets

$$\mathcal{S}_i^{+k} = \{s \in \mathcal{S} : o(s) = i,\ k \in Y(s)\}, \tag{10}$$

$$\mathcal{S}_i^{-k} = \{s \in \mathcal{S} : o(s) = i,\ k \notin Y(s)\}, \tag{11}$$

and require $|\mathcal{S}_i^{+k}|, |\mathcal{S}_i^{-k}| \geq 5$. No formal statistical test is applied; the threshold guards against degenerate probability estimates but does not bound false-positive edges under small samples. The empirical ownership retention gap is

$$\Delta_{ik} = \widehat{\Pr}[\hat{o}{=}i \mid o{=}i,\ k \in Y(s)] - \widehat{\Pr}[\hat{o}{=}i \mid o{=}i,\ k \notin Y(s)], \tag{12}$$

$$\mu_{k \to i} = N_i\,|\Delta_{ik}| \quad \text{if } \Delta_{ik} < 0, \tag{13}$$

and undefined otherwise. When $\Delta_{ik} < 0$, co-occurrence of k is associated with a lower rate of correct ownership of i. Δ_{ik} therefore measures *co-occurrence-induced degradation* of class i ownership in the presence of class k. It is a proxy for interference under polyphony, not a confirmed transfer of predicted ownership specifically to class k. The edge weight $\mu_{k \to i} = N_i\,|\Delta_{ik}|$ rescales the observed gap from $\mathcal{S}_i^{+k}$ (clips where k and i co-occur) to all i-owned clips, under the assumption that the interference pattern generalizes uniformly. This extrapolation treats the observed gap as representative; it does not measure the effect directly across the full population. Scaling by $|\mathcal{S}_i^{+k}|$ instead would restrict the analysis to clips where k is present. We keep the N_i form for consistency with Figure 2. Two structural limitations apply. First, because ownership is assigned to the longest-duration class, brief events such as *Dishes* rarely own clips when a continuous background class co-occurs, so short classes appear primarily as stealers rather than victims. Second, $\mathcal{S}_i^{-k}$ may include monophonic clips (only class i active), which are easier to own correctly than the polyphonic clips in $\mathcal{S}_i^{+k}$; Δ_{ik} therefore conflates class-k-specific interference with the general difficulty of polyphonic audio. The mass transfer graph is the weighted digraph $G = (V, E, w)$ with

$$V = \mathcal{Y}, \tag{14}$$

$$E = \{(k, i) \in V \times V : k \neq i,\ \Delta_{ik} < 0\}, \tag{15}$$

$$w(k, i) = \mu_{k \to i}. \tag{16}$$

An arc $k \to i$ marks that co-occurrence of k is associated with degraded ownership retention of i (interferer $\to$ victim). The "mass transfer" label is metaphorical: the graph measures associative

retention suppression, not confirmed directed ownership capture. Node dominance and vulnerability are the weighted out- and in-sums

$$D_i = \sum_{(i,j)\in E} w(i,j), \qquad U_i = \sum_{(k,i)\in E} w(k,i). \tag{17}$$

In figures, node color encodes D_i (source / stealer strength), node size encodes U_i (sink / victim strength), and edge width encodes $w(k,i)$. Large D_i indicates high outgoing edge weight; large U_i indicates high incoming edge weight. Intuitively, each clip is owned by its longest-duration class. An arc $k \to i$ means that when k co-occurs in an i-owned clip, the model more often loses i as predicted owner.