

REVE: EFFICIENT HALLUCINATION CORRECTION FOR LARGE AUDIO-LANGUAGE MODELS VIA REUSED ENCODER STATES

Hongjin Song, Jiasheng Kuang, Xinyu Yang, Qiuyu Fang, Ziyu Wu, Guowu Tan, Xiang Xie*

Beijing Institute of Technology, Zhuhai, Guangdong, China

*Corresponding author: Xiang Xie

ABSTRACT

Large audio-language models may mention acoustic events that are absent from the input. A separate audio event detector can verify these mentions, but doing so requires a second audio encoder and a separate forward pass. We propose Reused Encoder States for Verifying Events (REVE), a lightweight method that uses states already computed by the target model. One readout summarizes class scores across audio frames, while another uses pooled states from four consecutive frame intervals. Class-aware score fusion combines their outputs to verify generated event mentions without encoding the audio again. On AudioSet, REVE removes 92.9% of label-unsupported mentions under a faithful-mention recall constraint. With fewer added parameters and no second audio-encoding pass, REVE achieves a reduction comparable to those of CED-Tiny and CED-Base. Its complete verification latency is about 1/18 of the CED-Base path. Results on controlled DESED mixtures and different target-model architectures further confirm the effectiveness of encoder-state reuse.

Index Terms— large audio-language models, hallucination, representation probing, inference-time correction

1. INTRODUCTION

Large audio-language models (LALMs) usually connect an audio encoder and a large language model through a modality projector. This design supports tasks such as audio question answering and open-ended audio captioning. For example, Qwen2-Audio [1] first uses a Whisper-based audio encoder [2] to extract acoustic representations. A modality projector then maps these representations into audio embeddings that are compatible with the input space of the language model. The language model combines the audio embeddings with a text instruction to generate an answer or caption. Despite their strong audio understanding, LALMs may still describe acoustic events that are absent from the input. For example, a model may mention thunder in a rain-only clip or music in a speech recording. Such descriptions are often plausible. It is therefore difficult to verify them using only the generated text or token probabilities. Prior work reduces audio hallucinations through audio-aware decoding [3], activation steering [4], preference alignment [5], and broader evaluation frameworks [6]. However, caption hallucinations often occur at the event-mention level. A caption-level score cannot locate the exact unsupported text span. Changes to model training or decoding may also be difficult to apply to a deployed model.

A direct post-hoc solution first extracts event mentions from the generated caption, maps them to classes in an audio event ontology, and then queries a separate detector such as CED [7]. This process provides class-level acoustic evidence and links each detection result to a specific caption span. It can therefore retain supported event

mentions and remove unsupported ones. However, CED-Base introduces a high parameter cost and requires a second audio-encoder forward pass. CED-Tiny substantially reduces the detector size, but it still extracts acoustic features and encodes the waveform again. This additional path increases verification latency. Since the target model has already encoded the audio before generating the caption, we ask whether its existing encoder representations can provide event-presence evidence without running another audio model.

To this end, we propose **Reused Encoder States for Verifying Events (REVE)**. It uses two readouts: one summarizes frame-level class scores from the pre-projector states, and the other reads out the means of four contiguous, equal-length frame intervals. A class-aware calibrator fuses the two scores and produces the final event-presence score for each ontology class. REVE maps event spans in the generated caption to ontology classes and uses these scores to verify the corresponding events.

Our contributions are threefold. First, we formulate post-hoc audio caption correction as ontology-aligned event verification and introduce a two-scale verifier that reuses encoder states. Second, we evaluate hallucination reduction under a faithful-mention recall constraint and report the added computation and storage costs of the verification module. Third, source-disjoint controlled DESED experiments and cross-model experiments show that REVE generalizes across acoustic domains and LALM architectures.

2. METHOD

Fig. 1 shows the full caption generation and event verification process. REVE reuses frame-level encoder states before the modality projector as acoustic evidence. It maps event spans in the generated caption to an audio event ontology and retains only mentions supported by the corresponding class scores.

2.1. Encoder State Reuse

Let x denote the input waveform and q the captioning instruction. Let E , P , and G denote the audio encoder, modality projector, and language-model decoder of the target LALM, respectively. The audio encoder first extracts frame-level states from a log-Mel spectrogram:

$$H = E(\text{LogMel}(x)) = [h_1, \dots, h_T]^T \in \mathbb{R}^{T \times d}. \quad (1)$$

where T is the number of encoder frames, d is the dimension of each encoder state, and $h_t \in \mathbb{R}^d$ is the state at frame t . The modality projector then converts H into audio tokens that are compatible with the language model:

$$Z = P(H), \quad Z \in \mathbb{R}^{L \times d_G}. \quad (2)$$

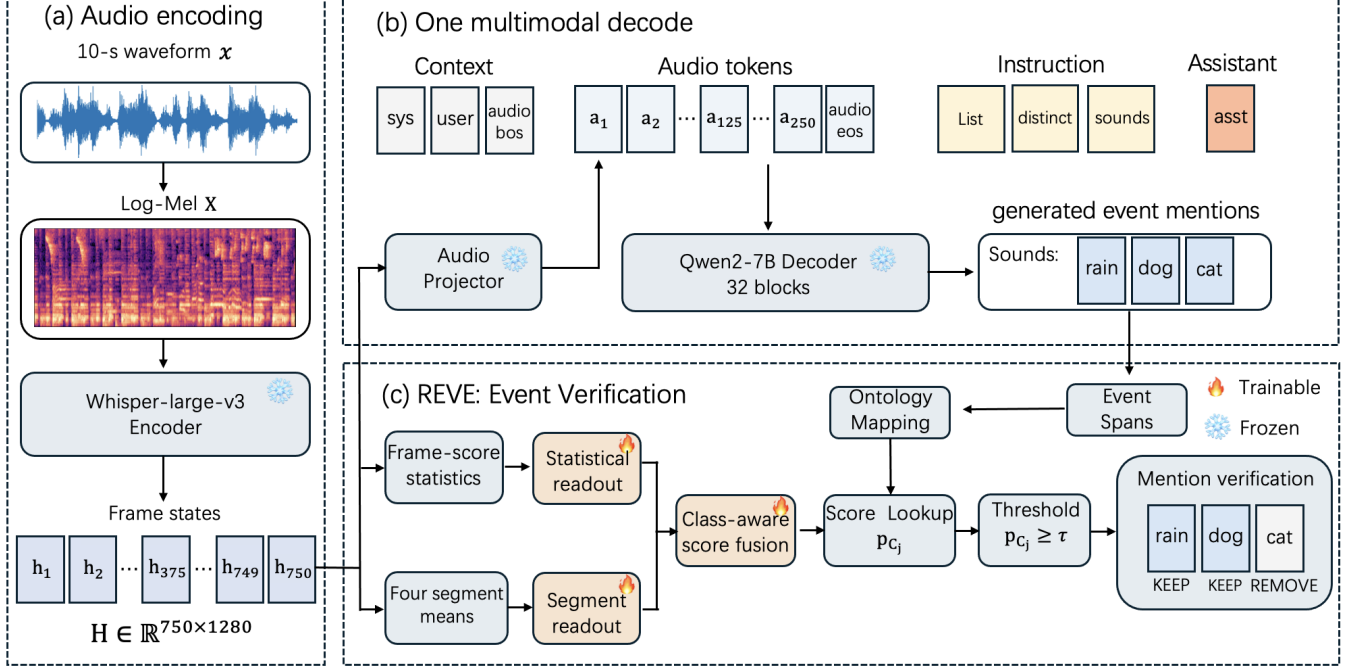


Fig. 1. Caption generation and event verification. (a) The audio encoder extracts frame-level states H . (b) The target LALM performs one standard multimodal decoding pass. (c) REVE combines frame-level score statistics with four temporal segment representations, fuses the two signals in a class-aware manner, and removes unsupported event mentions.

where L is the number of audio tokens and d_G is the token dimension. The language model generates caption C from Z and instruction q :

$$C = G(Z, q). \quad (3)$$

As shown in Fig. 1(a)–(b), the original caption generation path and REVE share the same encoder states H . REVE can therefore extract acoustic evidence directly from the existing states without reloading or re-encoding the input audio.

2.2. Ontology Mapping

The language model uses free-form event descriptions, whereas the readouts use fixed classes. REVE derives matching forms from the official AudioSet class names by normalizing case and separators and removing parenthetical qualifiers. Whole-word matching identifies event spans in caption C :

$$\mathcal{M}(C) = \{(s_j, c_j)\}_{j=1}^J, \quad (4)$$

where s_j is an event span and $c_j \in \{1, \dots, K\}$ is its ontology class. A matched node maps directly to a readout class or to its nearest represented ancestor. Unmapped spans are not edited. Reference labels use the same mapping.

2.3. Two-Scale Event Scoring

Mean pooling gives a compact clip-level representation but may weaken evidence for short events. To capture both global score distributions and local temporal information, REVE combines frame-level score statistics with four contiguous, equal-length temporal

segments. We first mean-pool all encoder frames to obtain the base clip representation:

$$e = \frac{1}{T} \sum_{t=1}^T h_t, \quad e \in \mathbb{R}^d. \quad (5)$$

A multi-label readout defines the base class logit $g_c = w_c^\top e + b_c$ and the frame-level logit $a_{t,c} = w_c^\top h_t + b_c$. For each mentioned class, we summarize the temporal distribution of its frame-level scores as

$$q_c = [g_c, \text{std}_t a_{t,c}, \max_t a_{t,c}], \quad (6)$$

and obtain $s_c^{\text{stat}} = \sigma(\alpha_c^\top q_c + \beta_c)$. In parallel, we divide the T encoder frames into four contiguous temporal segments $\mathcal{I}_l$ and retain the mean representation of each segment:

$$e_l = |\mathcal{I}_l|^{-1} \sum_{t \in \mathcal{I}_l} h_t, \quad l = 1, \dots, 4, \quad (7)$$

$$s_c^{\text{seg}} = \sigma(W_4[e_1; \dots; e_4] + b_4). \quad (8)$$

Both readouts are trained on AudioSet-balanced-train with class-balanced binary cross-entropy. Let $\ell(\cdot)$ denote the logit function and define $z_c = [\ell(s_c^{\text{stat}}), \ell(s_c^{\text{seg}})]^\top$. The final event-presence score is

$$p_c = \sigma\left(\theta^\top z_c + \delta_c + \gamma_c \frac{\mathbf{1}^\top z_c}{2} + b\right). \quad (9)$$

Here, θ and b are shared across classes, while δ_c and γ_c are the class-specific offset and slope. This regularized class-aware calibrator adds only 1,057 coefficients. All parameters of the target LALM remain frozen during training. The original readout that uses only one mean-pooled representation serves as the *Mean Readout* ablation.

2.4. Event Mention Correction

After ontology mapping, REVE scores only the classes mentioned in the caption. It produces score p_{c_j} for each event span s_j . This candidate-sparse implementation is equivalent to scoring all K classes. The threshold decision is defined as

$$r_j(\tau) = \mathbb{I}[p_{c_j} \geq \tau]. \quad (10)$$

where $\mathbb{I}[\cdot]$ is the indicator function and τ is the event-presence threshold. The event span is retained when $r_j = 1$ and removed when $r_j = 0$. The corrected caption is

$$C' = \text{Edit}(C, \{s_j \mid r_j(\tau) = 0\}). \quad (11)$$

REVE edits only event spans with an ontology class score. All other text remains unchanged.

3. EXPERIMENTAL SETUP

3.1. Datasets and Models

AudioSet. We train the two readouts on AudioSet-balanced-train [8]. For evaluation, we divide 1,000 AudioSet-eval clips into non-overlapping development and test sets of 500 clips each. Only the development set is used to select the class-aware calibrator and threshold. All methods use the same captions, event spans, ontology mappings, and reference labels.

DESED mixtures. We create 1,000 ten-second mixtures from isolated DESED foreground events [9]. They cover eight event families and contain one to three distinct events without background noise. Development and test sets contain 500 mixtures each and use disjoint source waveforms. The mixing records provide exact event-presence labels.

Models. Our main target is Qwen2-Audio-7B-Instruct [1]. For each ten-second clip, its encoder produces $H \in \mathbb{R}^{750 \times 1280}$ and the projector produces 250 audio tokens. Captions use greedy decoding. We also test Qwen2.5-Omni-7B and SALMONN-13B [10, 11].

3.2. Baselines

We compare signals from generation confidence, internal target-model states, and external audio models. **Token Confidence** uses probabilities aligned with each event mention. **Decoder Probe** uses the last-layer language-model states. **Mean Readout** uses one mean-pooled encoder representation. External baselines are **CLAP** [12], **AST** [13], **PANN Cnn14** [14], and the **CED-Tiny** and **CED-Base** audio taggers [7]. These models reprocess the audio, whereas REVE reuses target-model states.

3.3. Evaluation Protocol

Let $\mathcal{I}$ be the set of evaluated event mentions in the original captions. Let $y_i \in \{0, 1\}$ indicate whether mention i is supported by the reference labels. For scoring method m and threshold τ , let $r_i^{(m)}(\tau) \in \{0, 1\}$ indicate whether the mention is retained after pruning. We define residual false-mention density and faithful-mention recall as

$$D_m(\tau) = \frac{\sum_{i \in \mathcal{I}} (1 - y_i) r_i^{(m)}(\tau)}{|\mathcal{I}|}, \quad (12)$$

$$R_m(\tau) = \frac{\sum_{i \in \mathcal{I}} y_i r_i^{(m)}(\tau)}{\sum_{i \in \mathcal{I}} y_i}. \quad (13)$$

D_m is the fraction of all original evaluated mentions that are false but remain after pruning. Lower values are better. R_m is the fraction of reference-supported mentions that remain after pruning.

We report relative hallucination reduction as

$$Q_m(\tau) = 1 - \frac{D_m(\tau)}{D_{\text{no pruning}}}. \quad (14)$$

The reduction directly measures the fraction of original false mentions that are removed. However, it must be reported together with faithful-mention recall, because deleting every mention would otherwise give a 100% reduction. We therefore maximize the reduction under the development-set recall constraint below.

For each score-based method, we first obtain the set of development thresholds that satisfy the recall constraint:

$$\mathcal{F}_m = \{\tau \in \mathcal{T}_m : R_m^{\text{dev}}(\tau) \geq 0.75\}, \quad (15)$$

$$\tau_m^* = \max \mathcal{F}_m. \quad (16)$$

Class-balanced training makes the acoustic outputs decision scores rather than calibrated probabilities. We therefore select τ on the development set instead of fixing it at 0.5. REVE calibration and threshold selection use only grouped out-of-fold predictions from the development set. The complexity analysis includes only the added verification cost for a ten-second clip and excludes the shared target LALM. The data-processing, evaluation, and complexity-profiling scripts will be released on GitHub.

4. RESULTS

4.1. AudioSet Results

Table 1 reports results under the development-set recall constraint. REVE lowers residual density from 0.844 to 0.060 and removes 92.9% of false mentions at 0.756 recall. This is comparable to the 92.7% and 93.8% reductions of CED-Tiny and CED-Base, without a separate audio-encoder pass.

Table 1. AudioSet results at a development-set recall target of 0.75. REVE is shown in bold; underlining marks a best result only when it is achieved by another method.

Method	Residual		
	Recall $\uparrow$	density $\downarrow$	Reduction $\uparrow$
No Correction	1.000	0.844	0.0%
Token Confidence	<u>0.810</u>	0.563	33.3%
Decoder Probe	0.747	0.207	75.5%
Mean Readout	0.774	0.103	87.8%
CLAP	0.785	0.301	64.3%
PANN Cnn14	0.791	0.077	90.9%
AST	0.802	0.068	91.9%
CED-Tiny	0.786	0.061	92.7%
CED-Base	0.758	<u>0.052</u>	<u>93.8%</u>
REVE	0.756	0.060	92.9%

Table 2 compares the added deployment cost with CED. CED-Tiny reduces CED-Base from 85.71M to 5.50M parameters and from 10.32 to 8.27 ms complete latency. Both still extract features and encode the waveform again. REVE adds 3.38M parameters. Its GPU computation takes 0.29 ms and its synchronized complete latency is 0.57 ms, about 1/18 of the CED-Base path.

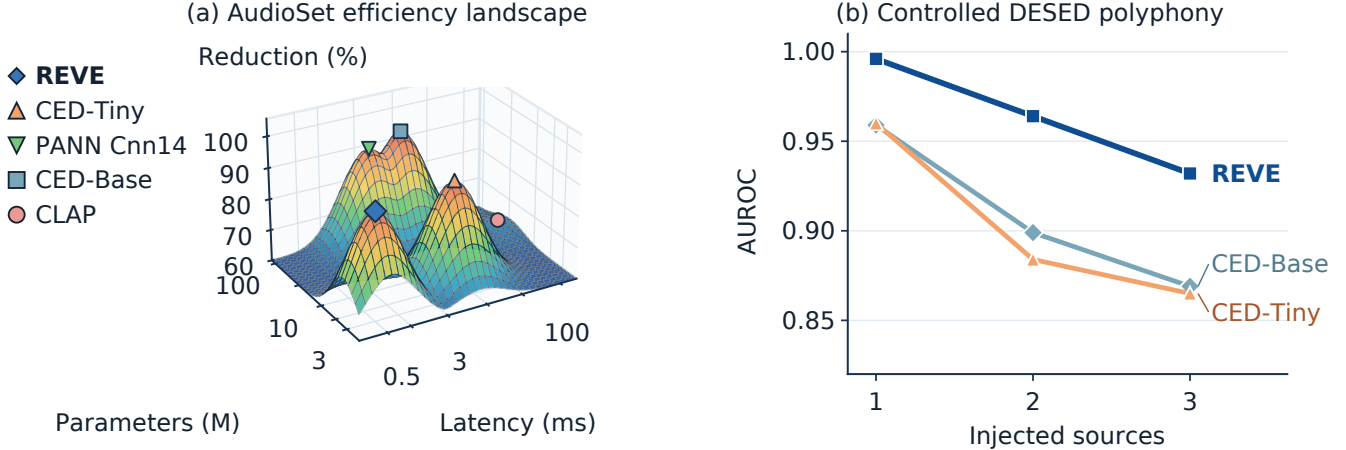


Fig. 2. Efficiency and controlled-mixture results. (a) AudioSet reduction versus complete added latency and parameters (log scales). Markers are measurements; the smoothed surface is a visual guide. (b) AUROC across controlled DESED polyphony levels.

Table 2. Incremental cost for a 10-s clip on an NVIDIA L20Y in FP32. GPU forward uses CUDA-event timing; CED complete latency also includes feature extraction and transfer. REVE is shown in bold.

Method	Parameters	GPU forward (ms)	Complete path (ms)
CED-Tiny	5.50M	4.20	8.27
CED-Base	85.71M	5.46	10.32
REVE	3.38M	0.29	0.57

Fig. 2(a) shows that REVE reaches a 92.9% reduction with only 0.57 ms of added latency, while PANN, CED, and CLAP are slower.

4.2. Ablation Study

Table 3 shows that the two readouts provide complementary evidence. Shared fusion raises mention AUROC from about 0.91 to 0.931 and lowers residual density to 0.082. Class-aware calibration keeps AUROC nearly unchanged but lowers residual density to 0.060, which supports more accurate decisions with one threshold across classes.

Table 3. Component ablation on AudioSet. Residual density is measured at the development-selected recall target of 0.75. REVE is shown in bold; underlining marks a best result only when it is achieved by another variant.

Variant	Mention AUROC $\uparrow$	Residual density $\downarrow$
Mean Readout	0.905	0.103
Statistics only	0.907	0.095
Segments only	0.909	0.095
Shared fusion	<u>0.931</u>	0.082
REVE	0.930	0.060

4.3. Controlled DESED Results

We use controlled DESED mixtures to avoid the missing-label ambiguity of AudioSet. The mixing records give exact presence labels for all eight event families.

Table 4 reports 0.959 AUROC and 0.915 AP for REVE. The strongest external detector, CED-Base, reaches 0.901/0.791.

Table 4. Source-presence diagnostic on controlled DESED mixtures. REVE is shown in bold.

Verification method	Micro-AUROC $\uparrow$	Micro-AP $\uparrow$
CLAP	0.821	0.644
PANN Cnn14	0.847	0.707
AST	0.887	0.761
CED-Tiny	0.896	0.779
CED-Base	0.901	0.791
REVE (DESED)	0.959	0.915

In Fig. 2(b), REVE obtains AUROCs of 0.996, 0.964, and 0.932 for one to three sources. It remains above both CED variants at every source count.

4.4. Cross-Model Results

We also evaluate REVE on Qwen2.5-Omni-7B and SALMONN-13B [10, 11] to test different target-model architectures.

Table 5. Complete REVE across target LALMs at the development recall target of 0.75.

Target LALM	Mention AUROC $\uparrow$	Density before $\rightarrow$ after	Reduction
Qwen2-Audio-7B	0.930	0.844 $\rightarrow$ 0.060	92.9%
Qwen2.5-Omni-7B	0.865	0.797 $\rightarrow$ 0.158	80.1%
SALMONN-13B	0.841	0.804 $\rightarrow$ 0.207	74.3%

REVE reduces residual density to 0.158 on Qwen2.5-Omni and 0.207 on SALMONN, or by 80.1% and 74.3%, respectively.

5. CONCLUSION

REVE corrects LALM captions using encoder states already available after caption generation. Its statistical and segment readouts remove 92.9% of false mentions under the recall constraint, matching CED-Tiny and CED-Base. REVE uses fewer parameters, avoids another audio encoding pass, and adds about 1/18 of CED-Base’s complete verification latency. DESED and cross-model results confirm the effectiveness of encoder-state reuse.

6. REFERENCES

- [1] Y. Chu, J. Xu, Q. Yang, H. Wei, X. Wei, Z. Guo, Y. Leng, Y. Lv, J. He, J. Lin, C. Zhou, and J. Zhou, “Qwen2-Audio technical report,” arXiv preprint arXiv:2407.10759, 2024.
- [2] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” in *Proc. ICML*, 2023.
- [3] T.-W. Hsu, K.-H. Lu, C.-H. Chiang, and H.-Y. Lee, “Reducing object hallucination in large audio-language models via audio-aware decoding,” in *Proc. IEEE ASRU*, 2025.
- [4] T.-E. Lin, K.-Y. Lee, and H.-Y. Lee, “Adaptive vector steering: A training-free, layer-wise intervention for hallucination mitigation in large audio and multimodal models,” arXiv preprint arXiv:2510.12851, 2025.
- [5] Y. Chen, W. Zhu, X. Chen, Z. Wang, X. Li, P. Qiu, H. Wang, X. Dong, Y. Xiong, A. Schneider, Y. Nevmyvaka, and Y. Wang, “AHA: Aligning large audio-language models for reasoning hallucinations via counterfactual hard negatives,” in *Findings of ACL*, 2026, pp. 29 294–29 306.
- [6] F. Zhao, Y. Chen, W. Lu, D. Zhang, X. Yue, and J. Wei, “Hallu-Audio: A comprehensive benchmark for hallucination detection in large audio-language models,” in *Proc. ACL*, 2026, pp. 38 797–38 816.
- [7] H. Dinkel, Y. Wang, Z. Yan, J. Zhang, and Y. Wang, “CED: Consistent ensemble distillation for audio tagging,” in *Proc. ICASSP*, 2024, pp. 291–295.
- [8] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, “AudioSet: An ontology and human-labeled dataset for audio events,” in *Proc. ICASSP*, 2017, pp. 776–780.
- [9] N. Turpault, R. Serizel, S. Wisdom, H. Erdogan, J. R. Hershey, E. Fonseca, P. Seetharaman, and J. Salamon, “Sound event detection and separation: A benchmark on DESED synthetic soundscapes,” in *Proc. ICASSP*, 2021, pp. 840–844.
- [10] J. Xu, Z. Guo, J. He, H. Hu, T. He, S. Bai, K. Chen, J. Wang, Y. Fan, K. Dang, B. Zhang, X. Wang, Y. Chu, and J. Lin, “Qwen2.5-Omni technical report,” arXiv preprint arXiv:2503.20215, 2025.
- [11] C. Tang, W. Yu, G. Sun, X. Chen, T. Tan, W. Li, L. Lu, Z. Ma, and C. Zhang, “SALMONN: Towards generic hearing abilities for large language models,” in *Proc. ICLR*, 2024.
- [12] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, “Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation,” in *Proc. ICASSP*, 2023, pp. 1–5.
- [13] Y. Gong, Y.-A. Chung, and J. Glass, “AST: Audio spectrogram transformer,” in *Proc. Interspeech*, 2021, pp. 571–575.
- [14] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, “PANNs: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 2880–2894, 2020.